Coventry University

DOCTOR OF PHILOSOPHY

**Mobile application of Artificial Intelligence to vital signs monitoring
multi parametric, user adaptable model for ubiquitous well-being monitoring**

Lewandowski, Jacek

Award date:
2014

Awarding institution:
Coventry University

Link to publication

Mobile application of Artificial Intelligence to vital signs monitoring: Multi-parametric, user-adaptable model for ubiquitous well-being monitoring

By

Jacek Lewandowski

Ph.D. August 2014

Coventry University

Mobile application of Artificial Intelligence to vital signs monitoring: Multi-parametric, user-adaptable model for ubiquitous well-being monitoring

By

Jacek Lewandowski

Ph.D. August 2014

A thesis submitted in partial fulfilment of the University's requirements for the Degree of Doctor of Philosophy

Abstract

Over the next decade, current reactive medicine, is expected to be replaced by a medicine increasingly focused on wellness so called personalised, predictive, preventive, and participatory (P4) medicine will be less expensive, yet more accurate and effective. It has been reported in the recent literature that wireless medical telemetry has the potential to contribute to this. However, research shows that current level of adoption of wireless medical telemetry in nearly every country is minimal. In addition to the technological challenges, other factors that seem to affect users' perception and acceptance of the new systems could generally be categorised as: cost efficiency, accuracy and effectiveness of monitoring, as well as security of the services.

The ultimate goal of this research is to demonstrate that ubiquitous vital signs monitoring of multiple parameters with patient-specific and adaptable inference model can make an accurate individualised prognosis of a patient's health status and its deterioration. With the ubiquities and individualised approach to the patient as opposed to the occasional, conventional population-based diagnostic flows, we could provide more accurate, cost efficient and effective solution, in order to answer many population-based problems of modern health care systems.

The framework developed by this research addresses elements that are common to current mobile health monitoring systems such as wireless sensing, data filtering and processing, as well as interconnection of external services, amongst others, but taken into a new integrative level. The model eliminates redundant tasks and therefore reduce cost, time and effort when developing Smart Wearable Systems (SWS) applications and minimising their "time-to-market". Having a framework such as the one proposed here will allow researchers and developers to focus more on the knowledge intrinsic to the patient-relevant data being collected and analysed, as opposed to technical developments and specific programming details.

The presented model enables adaptive monitoring of patients using patient specific models. This provides a more effective approach to identifying potential health risks and specific clinical symptoms of an individual, particularly when compared to the conventional, population-based diagnostic approaches currently used. It is foreseen that acquisition and analysis of multiple vital sign parameters from a single patient in real time, together with continuous adaptation of the level of detail of their analysis will enable a more precise understanding of the patient's health status and eventual diagnosis goals in the future.

By designing this highly adaptable, distributed model, capable of self-adjusting over time based on historical vital sign measurements, individuals are able to keep physically active, detection and early notification of potential illness risks is improved, and more accurate treatments "on-the-go" is possible with admission to hospital being reduced. The proposed solution is expected to continue to support illness prevention and early detection, enabling management of wellness rather than illness.

Acknowledgments

First of all I would like to thank my Director of Study dr. Alexeis Garcia-Perez for his continued support and motivation towards the end of this project. Secondly I would like to thank my former Director of Study dr. Hisbel E. Arochena for her inspiration to undertake this project as well as continued support throughout the undertaking of this project.

This work would have not been possible without the support of Coventry University Staff, in particular thank to my supervisory team: Professor Kuo-Ming Chao, Dr. Siraj Ahmed Shaikh as well as Professor Raouf N.G. Naguib, my former supervisor.

Special thanks to my parents and sister for their continued guidance and support. Finally, thanks to Agata, for her patience and understanding whilst this research and the thesis were completed.

List of Publications

1. Lewandowski, J., Arochena, H., Naguib, R., Chao, K. and Garcia-Perez, A. (2014) 'Logic-Centred Architecture for Ubiquitous Health Monitoring.' IEEE Journal of Biomedical and Health Informatics (accepted for publication in September issue)

2. Lewandowski, J., Salako, A. O. and Garcia-Perez, A. (2013) 'Saas Enterprise Resource Planning Systems: Challenges of Their Adoption in Smes.' e-Business Engineering (ICEBE), 2013 IEEE 10th International Conference on. 11-13 Sept. 2013

   ➢ *By August 2014 this publication has been cited by: 2*

3. Hinoveanu, L., Lewandowski, J., Fei, X., Arochena, H., Kandaswamy, P. and Dai, Z. (2013) 'Energy-Efficient Posture Classification with Filtered Sensed Data from a Single 3-Axis Accelerometer Deployed in WSN.' SENSORCOMM 2013, The Seventh International Conference on Sensor Technologies and Applications Held at Barcelona, Spain

4. Lewandowski, J., Arochena, H. E., Naguib, R. N. G. and Chao, K. (2012) 'A Simple Real-Time QRS Detection Algorithm Utilizing Curve-Length Concept with Combined Adaptive Threshold for Electrocardiogram Signal Classification.' TENCON 2012 - 2012 IEEE Region 10 Conference. 19-22 Nov. 2012

   ➢ *By August 2014 this publication has been cited by: 6*

5. Lewandowski, J., Arochena, H. E., Naguib, R. N. G. and Kuo-Ming, C. (2011) 'A Portable Framework Design to Support User Context Aware Augmented Reality Applications.' 3rd IEEE International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES), . 4-6 May 2011

   ➢ *By August 2014 this publication has been cited by: 10*

6. Lewandowski, J. and Arochena, H. E. (2011) 'Mobile Attendance and Time Monitoring System for M-Learning Applications: Design and Pilot Results ' In Sánchez, I. A. and Isaías, P. (ed.) IADIS International Conference Mobile Learning 2011. IADIS. Available from <http://www.iadisportal.org/digital-library/mdownload/mobile-attendance-and-time-monitoring-system-for-m-learning-applications-design-and-pilot-results>

7. Lewandowski, J., Arochena, H. E. and Naguib, R. N. G. (2009) 'Attendance and Time Monitoring of Day-Clinic Patients Using Wireless Communication.' 4th IEEE International Conference Humanoid, Nanotechnology, Information Technology Communication and Control, Environment and Management (HNICEM). Held at Manila, Philippines

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ABP   – Arterial Blood Pressure

AC   – Alternating Current

ACCEL   – Accelerometer

ADC   – Analog-To-Digital Converter

AF   – Adaptation Function

AFE   – Analog Front End

AI   – Artificial Intelligence

ALMA   – Architecture Level Modifiability Analyses

AnEx   – Analog Expansion Board

ANN   – Artificial Neural Networks

ATAM   – Architecture Trade-Off Analyses Method

BAN   – Body Area Networks

BMU   – Best Matching Unit

BP   – Blood Pressure

BW   – Baseline Wander

CLDC   – Connected Limited Device Configuration

DB   – Diastolic Blood Pressure

DC   – Direct Current

DCE   – Data Circuit-terminating Equipment

DTE   – Data Terminal Equipment

DWT   – Discrete Wavelet Transformation

ECG   – Electrocardiogram

EF   – Expert Function

EMG   – Electromyography signal

EWS   – Early Warning Score

FD   – Failed Detection

FIR   – Finite Impulse Response

FN   – False-Negative

FP   – False-Positive

FTSP   – Flooding Time Synchronization Protocol

HAA   – Hardware Abstraction Architecture

HAL   – Hardware Abstraction Layer

HR   – Heart Rate

IF   – Integration Function

IIR   – Infinite Impulse Response

jMENN   – Joone Mobile Edition Neural Network

JOONE   – Java Object Oriented Engine

LMS   – Least Mean Squares

LWUIT   – Lightweight User Interface Toolkit

MAP   – Mean Arterial Pressure

MEMS   – Method for Evaluating Middleware Architectures

MEWS   – Modified Early Warning Score

MIMIC   – Multi-parameter Intelligent Monitoring for Intensive Care

NAN   – Near-me Area Networks

NEWS   – National Early Warning Score

NLMS   – Normalised Least Mean Squares

OPAMP   – Operational Amplifier

+P   – Positive Predictive Value

PAN   – Personal Area Networks

PCA   – Principal Component Analysis

PDA   – Personal Digital Assistant

PHA   – Personal Health Assistants

PHS   – Personal Health System

PS   – Personal Server

PTT   – Pulse Transit Time

PULSE   – Pulse rate

PVC   – Premature Ventricular Contractions

QP   – Quadratic Programming

RAG   – Red Amber Green Report System

RBF   – Radial Basis Function

RESP   – Respiratory Rate

RLS   – Recursive Least Squares

RMSE   – Random Mean Square Error

SAAM   – Software Architecture Analyses Method

SBP   – Systolic Blood Pressure

SMO   – Sequential Minimal Optimization

SNR   – Signal-To-Noise Ratio

SOM   – Self Organizing Maps

SpO_2 – Pulse Oximetry / Blood Oxygen Saturation

SVM   – Support Vector Machine

SWS   – Smart Wearable Systems

TASOM   – Time Adaptive Self-Organizing Map

TEMP   – Body Temperature

TF   – Tracing Function

TP   – True-Positive

WAN   – Wide Area Networks

WBAN   – Wireless Body Area Networks

WBASN – Wireless Body Area Sensors Network

WPAN   – Wireless Personal Area Networks

WSN   – Wireless Sensor Network

WSN   – Wireless Sensors Networks

Chapter

1

Introduction

Existing health care systems are mostly structured and optimised for reacting to crisis and managing illness. However, with convergence of new system approaches to disease management, with new measurement and visualisation technologies, as well as with new computational and mathematical tools, modern medicine is about to undergo a fundamental transition, which aims to transform the nature of healthcare from reactive to preventive. These changes are catalysed by new technologies, which allow focusing on prevention and early detection of disease and optimal maintenance of chronic conditions. Further advancements will eventually trigger the emergence of personalised medicine — a medicine that will focus on individual patients. Over the next decade, current reactive medicine, is expected to be replaced by a medicine increasingly focused on wellness, so called personalised, predictive, preventive, and participatory (P4) medicine that will be inexpensive, accurate and effective (Hood and Friend 2011) .

This radical change in healthcare provision is largely imposed by current economic, social, and demographic trends. According to the recent World Health Report published by the World Health Organization (World Health Organisation 2010) by 2050 well-developed countries are expected to face major challenges in the way current health care services are deployed and delivered. This is due to 1) an aging population, 2) increased life expectancy, and 3) population growth, amongst others. The United Nations report (United Nations 2012) states that this trend is global and, for instance, the more developed regions' population over 60 is expected to increase from 279 million in 2012 to 418 million in 2050, while at the same time its total population will remain largely unchanged at 1.3 billion. This will account for 32% of their population. These factors will have a significant impact on the future high-rising costs of healthcare liabilities. According to Martin et al. (Martin, Lassman, Whittle *et al.* 2011) only in 2009, U.S. health care spending grew by 4% to the level of $2.5 trillion reaching 17.6% of the Gross Domestic Product (GDP). This growth rate of health expenditures outpaced the growth of the overall economy and will continue to grow in the next years. In response to this we need more accurate, pre-hospital and prevention-oriented health care system, which will take care of a person's physical health status at its earliest stage, through physical activity management, status monitoring and assessment, as well as early notification in case of an emergency situation.

One of the approaches to the solution of these problems, studied by research communities in recent years, is the use of telecommunication and information technologies as means to provide clinical health care at a distance. This approach has been called telemedicine. Telemedicine can help eliminate distance barriers between a patient and the doctor, and can improve access to medical services that would often not be consistently available for instance in distant rural communities, or to save lives in critical care and emergency situations (Shnayder, Chen, Lornicz *et al.* 2005). Pervasive telemedicine, often referred to as m-health, in turn focus on provision of health services "on the go". Its applications include the use of mobile devices in collecting community and clinical health data, delivery of healthcare information to practitioners, researchers, and patients, real-time monitoring of patient vital signs or direct provision of care via mobile telemedicine. It contends to be the next generation of e-health systems, delivering user tailored health care services with the vision of empowered health care on the move (Istepanian, Laxminarayan and Pattichis 2006).

Building upon fast and steady advancements in wireless networking, mobile computing, microelectronics and sensor technologies, accurate ubiquitous health monitoring of a population by means of smart, non-intrusive, wearable sensor devices is an important trend of modern telemedicine, that when implemented well, could lead to proactive and individualised healthcare. Smart wearable systems (SWS) are defined as end-to-end, sensor-based integrated systems, capable of sensing, processing, and communicating medical data to interested parties, such as the medical professionals and emergency services, or store it for further reference. Sensor devices used in these systems can take different forms from implantable (Valdastri, Rossi, Menciassi *et al.* 2008, Halliday, Moulton, Wallace *et al.* 2012) and in-vivo devices (Rodrigues, Caldeira and Vaidya 2009, Pan and Wang 2012), through wearable sensors (Shnayder, Chen, Lornicz *et al.* 2005, Tia, Pesto, Selavo *et al.* 2008), to non-contact radar measurements (Morgan and Zierdt 2009, Changzhi, Cummings, Lam *et al.* 2009). Vital signs, most commonly subject to measurement, are physiological functions or indices such as: ECG, respiration, blood pressure, blood oxygen saturation (SpO_2), or temperature, amongst others. They present a potential to provide critical information that could enable the assessment of the long-term health status of an individual, as well as prevent and early diagnose any unsought health status changes.

Despite early thoughts, such as Otto's *et al.* (Otto, Milenkovic, Sanders *et al.* 2006), which emphasise potential and importance of SWS systems in transition to ubiquitous health care approach, yet their adoption in nearly every country is minimal. Questions remain open as to what form and properties systems should have in order to make healthcare really available to anyone, anytime, and anywhere. This is thought to have the potential to improve health in a cost-efficient manner and offer more accurate and effective service. Further research and developments that will answer this questions, should finally lead to a transition from medical systems for patients to health systems for citizens, a significant change from illness to wellness monitoring and management.

1.1 Challenges: Cost efficiency, accuracy and effectiveness of health monitoring

Naturally, transition to health systems for citizens will only be possible with the right support from technology, as well as by answering many crucial adoption challenges of modern m-health systems. Comprehensive list of issues that academic research on acceptance of Smart Wearable Systems must tackle, was compiled by Chan *et al.* (Chan, Estève, Fourniols *et al.* 2012) and include, amongst other issues:

- system accuracy, reliability, and unobtrusiveness (Hensel, Demiris and Courtney 2006, Ko, Lu, Srivastava *et al.* 2010)
- availability, integration and interoperability of services (Istepanian, Jovanov and Zhang 2004, Blobel 2007)
- cost efficiency, including development and maintenance costs (Bergmo 2010)
- technological capabilities to meet health care professionals and end-users requirements (Chan, Estève, Fourniols *et al.* 2012)
- effectiveness in ensuring medical, wellness and quality of life benefits (Patel, Park, Bonato *et al.* 2012)
- security, privacy, ethical and legal barriers (Dickens and Cook 2006, Kluge 2011)

Reviewing other literature on issues concerning adoption of SWS systems (Alemdar and Ersoy 2010, Atallah, Lo and Yang 2012), factors that mostly affect SWS perception and acceptance are described in general as: cost efficiency, accuracy, effectiveness, as well as security of the services.

1.1.1 Cost efficiency of service

As literature suggest the knowledge and understanding of the costs of telemedicine is still largely incomplete in evidence. However, analyses show that there is fair evidence of cost-effectiveness for many telehomecare applications, only few researchers draw firm conclusions. This is due to the heterogeneity among cost-effectiveness indicators in the applications and the methodological limitations of the studies what impede the possibility of generalising the findings (Ekeland, Bowes and Flottorp 2010).

Some researchers found economic benefits from the implementation of wearable and pervasive health care systems to have a considerable impact on the finances of societies with a shortage of personnel for taking care of the elderly or chronically ill (Robinson, Stroetmann and Stroetmann 2004). The evidence of cost effectiveness that has been found, suggests that it reduced use of hospitals, improved patient compliance, satisfaction and quality of life (Rojas and Gagnon 2008). The comparison of the costs of telemonitoring and usual care for heart failure patients (Seto 2008) found that telemonitoring could reduce travel time and hospital admissions. An example of such cost reduction present Raatikainen et al. (2008) in the report on the first experiences with the implantable cardioverter defibrillator (ICD) monitoring system in

Europe. Forty-one patients with ICD devices were selected and undergone 9 month trial, whereas in-office visits were substituted by remote data transmissions. As a result, remote monitoring diminished the direct and indirect annual costs of ICD follow-up by €524 per person what equals to 41% of total cost. While the total cost of standard follow-up among 41 participants accounted for up to €52k, those using remote monitoring estimated around €30k. However these results are promising and it is likely that remote monitoring would save a substantial amount of time and money, these results cannot be directly extrapolated on other cases due to uniqueness of each case and the complexity of the total cost calculation. It is also important to note that benefits if any are likely to be realised in the long term.

Other researchers such as Barlow et al. (Barlow, Singh, Bayer *et al.* 2007) who reviewed home telecare for older people and patients with chronic conditions or Deshpande and Mckibbon (Deshpande and Mckibbon 2008) who reviewed costs of synchronous telehealth in primary care, found neither satisfactory evidence nor consistent results about cost-effectiveness of telemonitoring. Instead, they identified particular cost related limitations of telemonitoring. They concerns refer to wider social and organisational costs of telemedicine: costs of maintaining health services of interventions as well as costs of time, effort and resources needed to develop new telemonitoring applications. It is difficult to find the robust analysis of these direct and indirect costs of wearable systems in literature. The most what can be found is the summary of system components costs, and claims that a regular widespread operation would shift their costs.

Reviewing current SWS systems, we can observe that these are usually single problem tailored solutions that sense, transmit and process single or few parameters only. Recent advances and popularisation of smartphone and new sensor devices have put such systems at the cross-road, demanding for more unified, more integrative, more intelligent and cognitive systems that can get easy and quickly assembled and deployed to solve particular research problems or answer commercial needs. Successful solution that would enable this, requires radical rethinking and redesign of telemedicine methodologies which will adopt current standards in networking, service infrastructure and cognitive systems development.

Telemedicine in its current form of telemonitoring, where system acts as a communication channel between medical staff and patients, becomes obsolete. The main requirement for most of those systems is to ensure that patients stay securely connected to their remote care professional. Because this approach does not reduce the health professionals involvement in the health monitoring and assessment process, the full cost reduction benefits cannot be observed. Even if some systems already implement intelligent data processing capabilities these are most often located on centralised servers rather than locally close to where data is acquired. As the data collection rate in pervasive healthcare systems is high the development of efficient data processing techniques is of great importance. In some cases 3-lead ECG may not be sufficient for identifying a cardiac disease or a single 3-axes accelerometer may not be capable of classifying all activities of the people. In these cases, more sensors that operate

simultaneously will be needed and the gathered data will increase (Alemdar and Ersoy 2010). Lack of decision support or only centralised processing in this case will not only affects the transmission capabilities, accuracy, response time or availability of the service in case of rural areas but what's most important the cost of the service maintenance.

As the amount of sensor devices will rise these create problems not only in terms of their interoperability but more importantly will put a pressure on developing easily deployable pervasive systems, where the cost reduction associated with systems' development is foreseen. Due to heterogeneity of medical as well as IT infrastructures and devices, the development and deployment process of new health monitoring systems requires every time a considerable amount of effort, cost and resources being spent on hardware and software." Currently a development of a single application requires the knowledge of diverse areas such as Electronics, Advanced Low Level Programming, Signal Processing, Control Theory, Networking, Mobile Application Development, Artificial Intelligence, Mobile GUI programming to name only technology specialists and omitting all domain specialists such as medical practitioners, sport scientists, etc. With such diversity of skills, ease and cost efficiency of deployment becomes a challenge to be considered.

Cost reduction benefits that could come out of more unified approach to development and more automated wellness monitoring systems, such as Personal Health Assistants (PHA), that require no or only little human intervention, are great. Therefore development methodologies with "short time-to-market" for new solutions/applications, with more code reusability and process-related application models that use third party measurement devices, are with no doubt the future of SWS Systems.

Complex data monitoring and integrated analysis of various parameters	• More accurate results could be achieved by taking into consideration aspects such as adaptability and environmental context (body, near-body & environmental sensing).
Lack of a unified, modularised Mobile Platform or Application Programming Interface (API) for biofeedback monitoring	• Lack of an API that could offer predefined methods and paradigms but still being fully customizable to operate with custom sensor nodes, which could enable rapid data aggregation to the mobile phone.
Extensive and often platform targeted implementation	• In order to get simple sensor readings, we are currently restricted by development restricted non-interoperable platforms
Lack of "on-the-go" intelligent data processing	• Real-time processing capabilities (such as ready to use ANN engine) needed.
Requirement for highly skilled IT specialists	• Development of a single application currently needs the knowledge of diverse areas such as: Electronics, Advanced Low Level Programming, Signal Processing, Control Theory, Networking, Mobile Application Development, Artificial Intelligence, Mobile GUI programming

Figure 1.1 Limitations of current SWS developments.

1.1.2   Accuracy of monitoring

As research shows users of SWS systems, especially elderly one, perceive independence and autonomy crucial for their everyday life, so that any systems or technology that can prolong that independence tends to be highly considered (Steele, Lo, Secombe *et al.* 2009). This highlights the need for a more accurate, ubiquities, pre-hospital and prevention oriented health care systems, which will take care of a person's physical health conditions at their earliest stage, through physical activity management, status monitoring, assessment as well as early notification in case of an emergency situation. This requirement poses serious challenges to new SWS systems in terms of how to organise the data and produce meaningful information that evolve into knowledge.

One of these key challenges is the autonomous ability to early diagnose and assess prognoses of developing pathologic conditions by an individual. The reality is that current medicine still do not fully understood causes of effects of many disease nor their processes. Even if diseases were fully understood, current healthcare provision does not fully take into account population variability when making individualised diagnosis, treatments, or prognosis, simply because long term monitoring data that could support such reasoning very often does not exist. Moreover, signs that might present themselves frequently during normal daily activities may disappear while the patient is hospitalised or undergoes examination; causing high costs, diagnostic difficulties and possible errors (Lymberis and Dittmar 2007).

Therefore disease models and treatments that guides clinicians, often base on statistical analyses of the population not individuals. Personalised health monitoring devices could be useful in early identification of medical conditions and facilitation of conventional clinical diagnosis processes by analysing disease relevant data and providing intelligent diagnostic assessment and alert feedback, either to the patient or directly to the healthcare professionals. Such integrated, intelligent and context aware health care provision could enable individuals to closely monitor changes in their own health status and maintain an optimal health status independently from the context of use and environment. As some early research shows, acquisition of multiple parameters and continuous adaptation of the interpretation depth could allow to more precisely follow the patient's health status and diagnosis goals (Alemzadeh,
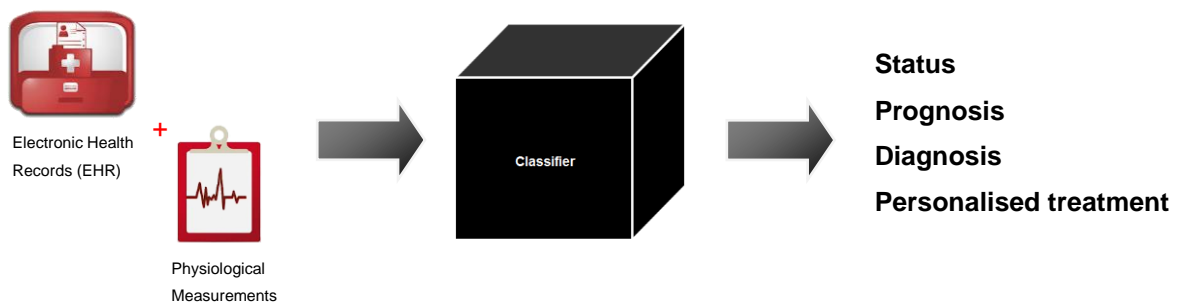
Figure 1.2 Machine learning in medicine.

Zhanpeng, Kalbarczyk *et al.* 2011). Therefore, robust machine learning algorithms and models are needed for self-learning, autonomous systems replacing rule based and static systems. Such user tailored and ubiquitous health care approach would further support prevention, early detection, and management of wellness rather than illness.

1.1.3   Effectiveness of interventions

Recently, questions are being asked about the effectiveness and efficiency of wearable health care systems. Despite large number of studies and systematic reviews on the effects of telemedicine, high quality evidence to inform policy decisions on how best to use telemedicine in health care is still lacking. Large studies with rigorous designs are needed to get better evidence on the effects of telemedicine interventions on health, satisfaction with care and costs. (Ekeland, Bowes and Flottorp 2010)

In order to fully realise potential of health and wellness monitoring through smart wearable technologies, researchers and providers have to work towards developing a comprehensive approach to health and wellness services. This is in contrast to what mostly take place nowadays, when researchers and providers focus is on devices and applications that monitor only single parameter or diseases. Because of complexity of health and well-being, which rarely depends on one parameter only, such approaches often miss the expectations of users as well as health providers. High accuracy autonomous diagnostic decisions can only be achieved through concurrent analysis and fusion of the results from multi-parameter signal analysis, including contextual and environmental data.

However integration of multiple parameters often sensed by third party sensor devices operating at different frequencies raises an interoperability problem. Despite attempt to establish standards of interoperable personal, connected health solutions neither through Continua Health Alliance (Carroll, Cnossen, Schnell *et al.* 2007), nor recently launched Bluetooth Smart or ANT+ (Evanczuk 2013) wireless technologies, seamless semantic integration of multiple third party sensor devices, still remains a problem. As communication between different devices occupies multiple bands and use different protocols it may cause interferences among different devices especially in the unlicensed Industrial, Scientific and Medical (ISM) radio bands.

Another problem exist at interconnection of SWS systems and disparate healthcare systems. Despite partial solutions have recently been introduced with mappings of HL7 Personal Health Monitoring Reports (PHMR) standard to improve interoperability with health professionals information systems (Yuksel and Dogac 2011), but only just some wearable prototypes have begun to use it. Therefore future pervasive healthcare systems must be designed with interoperability provisioning between different devices and services.

As many researchers suggest (Korhonen, Parkka and Van Gils 2003, Lymberis and Dittmar 2007, Swiatek, Stelmach, Prusiewicz *et al.* 2012) one way to overcome this problem

and achieve system effectiveness is through development of an integrated architectures for pervasive services with wearable systems and devices for home and on-the-go comfort, health and wellness. It is seen by many as the key point to maintain and boost SWS adoption, research and development. However found by Chan *et al.* (Chan, Estève, Fourniols *et al.* 2012) there is currently no smart wearable system on the market integrating several biosensors, intelligent processing and alerts to support medical applications. The need to combine sensors and services from different vendors, modalities and standards make it necessary to propose hardware independent software solutions for efficient integrated applications development.

All these issues not only prevent seamless medical data collection, increase the cost of the systems, and their upgrade capabilities, but they also limit the possibilities of making a shift to systems that are semantically interoperable, process-related, decision-supportive, context-sensitive, user oriented, and trustworthy (Blobel 2007).

1.2   A proposed solution: Multi-parametric, intelligent and user-adaptable monitoring

Classical remote telemonitoring system based on wireless sensor networks, typically consists of network of devices organised into three main layers: medical sensors, network, and application layers. There are numerous technical, functional and non-functional challenges in all three layers of wireless sensor networks, that affect accuracy, cost efficiency and effectiveness of health monitoring. As far as all issues should be addressed in order to fully enjoy the benefits of such systems, recent advances in smartphone technologies, leading to increased processing power, mobile operating systems and content-reach applications, point research and development efforts to application layer. Problems such as integration and interoperability aspects as well as new more accurate classification algorithms could be improved this way.
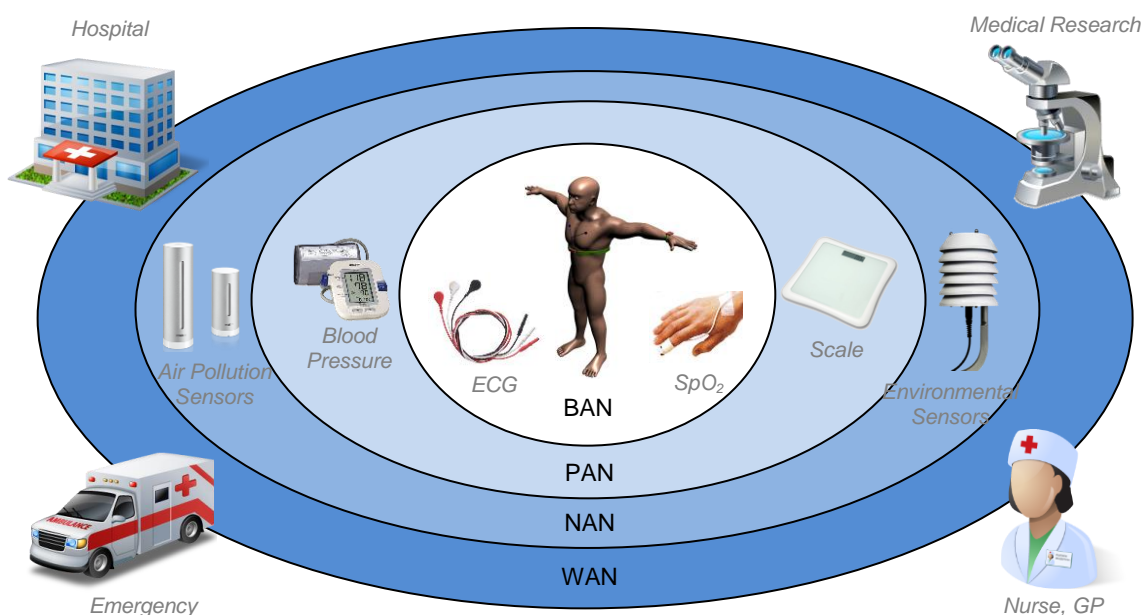
Figure 1.3 Modern ubiquities health and well being monitoring landscape.

Application layer, allows to isolate common behaviour that can be reused by several applications and to encapsulate it as system services. In this way multiple sensors and applications can be supported easily, through resource management and plug and play functions. With middleware, that can help to manage the inherent complexity and heterogeneity of medical sensors, the research can focus on the health data processing algorithms and their efficient development and deployment. Machine learning algorithms for healthcare, whether supervised, unsupervised or reinforcement, usually present a high level of complexity. Development and maintenance of such algorithms when applied to individual cases, on distributed and piecemeal systems, increase their intricacy by tenfold.

With far more flexibility in developing new mobile applications than ever before, the first part of this research focus on consolidation actions in order to propose a refined generic architecture of a multi-parametric, portable integrated, non-invasive framework for health monitoring and analyses. As part of these, system structure as well as environmental components and their relationships are determined, in order to propose more ubiquities and more personal data flow model that supports local artificial intelligence (AI) analysis. The prototype of this system is the smartphone based wireless sensing middleware with customisable wireless interfaces and plug'n'play capability to easily interconnect multiple third party sensor devices and services in Wireless Body (BAN), Personal (PAN), Near-me (NAN) and Wide (WAN) Area Networks landscape. Pivotal part of the platform is the integrated inference engine/AI runtime environment that allows the mobile device to serve as an intelligent, user-adaptable personal health assistant (PHA). One of the novelties of this solution is the algorithm's rapid visual development and remote deployment model. It is enabled by the complementary visual inference engine editor, that comes with the framework, allowing AI specialist along with medical experts to build data processing models by gluing together different components and controlling the application logic with scripts. The editor allows to instantly deploy such models remotely on patient mobile device. This approach shift focus from software and hardware development, to medical and health process implementation. Presented model helps to speed up development and deployment of medical and health decision processes what directly impact on response time and accuracy of medical interventions.

The framework is expected to addresses elements that are common for current mobile health monitoring systems such as wireless sensing, data filtering and processing, as well as interconnection of external services, amongst others, but taken into a new integrative level. This will eliminate redundant tasks and therefore reduce cost, time and effort when developing such applications in the future as well as minimising their "time-to-market". This, on the other hand, will also mean that the applications will be ready for deployment in a shorter period increasing the benefits for the end users. Having a framework such as the one proposed here will allow researchers and developers to focus more on the knowledge intrinsic to the data rather than on technical programming details.

The embedded machine learning capabilities of the framework are foundation for the second part of this research project, which focus on development of a multi-parametric intelligent model for accurate, user-adaptable assessment of well-being status base on collected vital signs. Vital signs are objective measures of physiological functions, which help assess the most basic body functions. However, as every human being is different, accepted ranges for each sign are recognised as a valid approximation only, which set an orientation point for further analysis. Current vital signs assessment mainly relies on comparing the measured values with medically approved normal ranges. An example of such generally accepted vital signs ranges for adults are the Modified Early Warning Scores (MEWS) (Subbe, Kruger, Rutherford *et al.* 2001) used in medical admission to hospitals and as a mean of periodic monitoring of in-patients. Obtained measurements however largely depend on age, gender, race, body weight, height, and/ or individual's exercise tolerance, amongst others. Consequently, when monitoring patient's optimal health status, more complex, context-aware, and patient-specific monitoring as well as integrated analysis, have to be used in order to give more accurate results and correctly identify abnormal values and their medical significance.

Solution proposed is the two-stages distributed event-triggered hybrid data processing model. The first stage consists of a distributed data filtering against medically approved ranges and events issuing in case thresholds are exceeded. The responsibility for stage one processing relay on sensor nodes. The second stage is the event-triggered integrated data fusion module which uses Kohonen Self Organizing Maps (SOM) and descriptive statistics such as measures of central tendency and spread, to undertake user tailored health status deterioration analyses base on collected vital signs measurements. The responsibility for stage two processing relay on personal server (PS), typically smartphone device. The prototype system integrates and continuously analyses five different vital signs recognised as standards in most medical settings. These are: Body Temperature (TEMP), Electrocardiogram (ECG), Blood Pressure (BP), Respiratory rate (RESP), and Pulse oximetry (SpO2).

The model is expected to enable patient specific and adaptive monitoring that will be more effective in identifying the potential health risks and specific clinical symptoms of an individual, compared with the conventional population-based diagnostic flows. Acquisition and analysis of multiple parameters in real time and continuous adaptation of the interpretation depth is envisaged to help more precisely follow the patient's health status and eventual diagnosis goals. By designing this highly adaptable, distributed model, that is capable of self-adjusting over time based on historical vital signs measurements, will help to keep subjects physically active, provide detection and early notification of potential illness risks, or facilitate more accurate treatments "on-the-go". Proposed solution is expected to continue to support prevention, early detection, and management of wellness rather than illness.
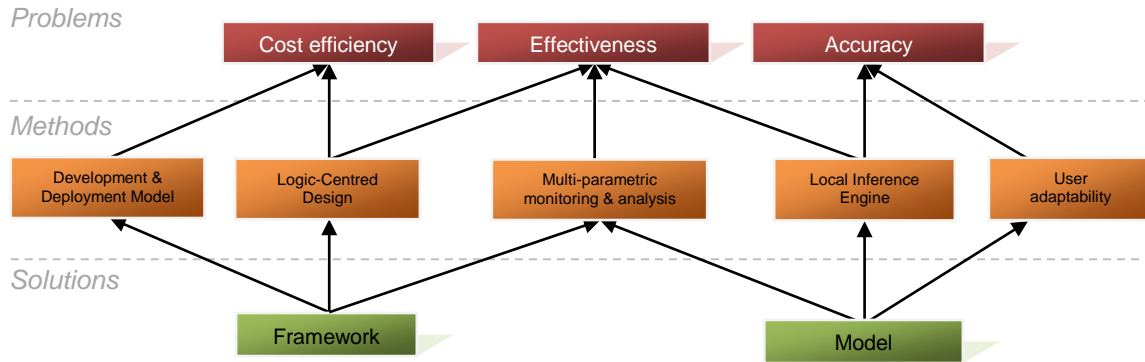
Figure 1.4 Mapping between problems, methods and solutions.

1.3 Research Aim and Challenges

The fundamental research challenge of this research project is to demonstrate that vital sign monitoring of an individual to a clinical standard can be achieved by using currently available mobile technology. As such, the monitoring fidelity achieved should provide a degree of confidence to allow medically informed decisions to be taken. To evaluate the proposed approach, the Intensive Care Unit (ICU) data has been used. This data is suitable as it:

1) monitors to a high clinical standard;
2) has a degree of automation that drives an alarm based system what allows our system to be benchmarked against another autonomous system;
3) is regularly used to inform medical decisions.

In this study the author seeks to determine if by using today's commodity devices the mobile monitoring and an alarm system can be build, that performs no worse than one found in an ICU department. The research challenges of this project are, therefore, threefold:

a) Miniaturise the monitoring infrastructure so it is portable in nature and can work on commodity computing devices (e.g., smart phone, tablet).
b) Ensure cost efficiency of monitoring infrastructure to ensure a practical, sustainable, solution is possible.
c) Allow the system to exhibit a greater degree in alarm confidence as no medical professional will be present to reset false alarms (which are generally high in ICU).

There are a multitude of monitoring devices available in the home marketplace that do monitor human activity, yet non acknowledges an ability to perform to a standard useful for medical diagnosis. Approach taken in this project is to employ a wide range of tools and techniques to allow such devices to become more accurate and suitable for clinical use, making a step towards transferring ICU quality monitoring into a commodity hardware and software solution.

11

1.3.1 Research objectives

Over the length of this project the author design and develop a ubiquities framework for patient-specific well-being monitoring and assessment of multiple vital signs collected over WSN. The proposed model is envisage to be cost-efficient, effective and accurate solution that supports individualised healthcare approach. Specific objectives of the research are to:

1) Propose an architecture for ubiquities physiological parameters analysis on mobile devices that determines main system/environment components, component relationships and overall structure of well-being analysis over WSN.

2) Model, build and integrate Inference Engine/AI runtime environment to portable devices to enable development of local Artificial Intelligence (AI) algorithms for health and well-being analyses.

3) Propose new rapid visual development and remote deployment model which will simplify and speed up the process of building remote data processing models.

4) Develop and test a prototype system for vital signs monitoring that implements a new multi-parametric, user adaptable and intelligent model for accurate assessment of human well-being status and eventually deterioration analyses of collected vital signs.

1.3.2 Hypothesis

The ultimate goal of this research is to demonstrate that ubiquitous vital signs monitoring of multiple parameters using commodity mobile devices, equipped with patient-specific and adaptable inference model, can make an accurate individualised prognosis of a patient health status and its deterioration. With the ubiquities and individualised approach to the patient as opposed to the occasional, conventional population-based diagnostic flows, we could ensure more accurate, cost efficient and effective solution that could help to answer many population based problems of modern health care systems. In order to prove it, two hypotheses were tested:

1) Development and deployment of concurrent ubiquitous monitoring and intelligent local analysis of multiple physiological signals on mobile devices with use of custom or third party sensing device can be effective and cost efficient.

2) Multi-parametric, patient specific and adaptable vital signs monitoring reduces the number of false alarms and produce results that are more accurate and effective than those of threshold based techniques used by conventional ICU health monitors.

In order to test the first hypotheses the proof of concept prototype framework and mobile inference engine were built, as part of this project. The second hypotheses was tested in the real data sets obtained from Physiobank database (Goldberger, Amaral, Glass *et al.* 2000). As the overall results show, high accuracy diagnostic decisions can be achieved by fusion of the

results from multi-parameter signal analysis delivered by cost effective and efficient patient-specific deployments.

1.3.3 Evaluation methodology

The objective of the evaluation methodology adopted in this study is twofold. The first objective aims to evaluate the middleware architecture of the proposed portable vital signs monitoring framework in light of multiple quality attributes. The second objective aims to evaluate whether method/enhancements proposed for data processing model yields an improved performance over the existing algorithm(s).

The evaluation of middleware architecture of the proposed portable vital signs monitoring framework, is done according to a Method for Evaluating Middleware Architectures (MEMS), proposed by Liu et al. (2006). MEMS is a scenario-based evaluation method dedicated to middleware architectures, which evaluate its multiple quality attributes, by leveraging generic qualitative and quantitative evaluation techniques such as prototyping, testing, rating, and analysis. It measures middleware architectures by rating multiple quality attributes with respect to key scenarios that describe the behaviour of the middleware in particular context. The evaluation process is driven by concerns about the quality attributes for specific designs using the middleware. Detailed discussion of the middleware evaluation methodologies can be found in Section 8.2.

The evaluation of data processing model, is done according to the statistical evaluation of experimental results. The evaluation start with the implicit hypothesis being made that the proposed method/enhancement yields an improved performance over the existing algorithm(s). To accept or reject the hypothesis a number of test data sets were selected for testing. The proposed algorithm were run and the quality of the resulting models was evaluated by calculating the number of True Positives (TP), True Negatives (TN), False Positive (FP) and False Negatives(FN) on each data set. In order to ensure comparability of proposed method with other algorithms, an appropriate performance measure(s) were used, such as: classification accuracy, sensitivity, specificity, positive predictive value, or failed detection. To compliment this evaluation and ensure that the difference between results of the proposed methods and those presented in the literature are non-random, a paired t-test was computed. It checks whether the average difference in performance of each pair of algorithms over the data set is significantly different from zero. T-test assume that each sample is drawn from a data set of normal or near-normal population distribution. Both numerical and statistical measures were defined according to evaluation standards proposed in the literature. These are discussed in detailed in Section 8.3 for each algorithm individually, due to different interpretation of results for multiple data sets.

1.4 Contributions

The main contribution to knowledge which is the result of this research project concerns wide range of tools and techniques to allow mobile health monitoring devices to become more accurate and suitable for clinical use, making a step towards transferring ICU quality monitoring into a commodity hardware and software solution. Other substantial research and development outcomes, which either enabled or are part of this contribution, include:

1) The ubiquitous architecture for physiological signals monitoring and analysis over WSN. The architecture creates a means for categorisation, determination and organisation of core components to simplify the development and use of WSN for physiological signals analysis. It consists of two elements:

    a) Sensor Node Middleware for TinyOS platform which consists of simple and lightweight components that sample, filter and process physiological signals on sensor node. The proposed middleware simplifies code reusability, application flexibility and customization of components with the traditional TinyOS development and deployment model.

    b) Personal Server Middleware, which provides integrated sensor nodes management, data aggregation, real-time data processing and transmission, as well as inference capabilities. It is a central component deployed on a smartphone device that links local WSN nodes and external WAN services.

2) The hardware design of two integrative vital signs monitoring devices:

    a) Chest strap to measure Heart Rate (HR), Electrocardiogram (ECG), Body Temperature (TEMP), and Respiratory Rate (RESP).

    b) Arm band to measure blood-oxygen saturation level (SpO_2) and pulse rate (PULSE).

    Combination of both devices allows to additionally measure the Mean Blood Pressure (MBP). All those vital signs are considered the standard for health status assessment in most medical settings.

3) The integrated inference engine/AI runtime environment that allows the mobile device to serve as a user-adaptable personal health assistant. The novelty of this system lays in a rapid visual development and remote deployment model. The complementary visual Inference Engine Editor that comes with the package, enables AI specialists, alongside with medical experts, to build data processing models by assembling different components and instantly deploying them (remotely) on patient mobile device.

4) A robust, high accuracy real-time QRS detector for noisy applications such as long-term ECG monitoring during normal daily activities. The algorithm exploits a modified curve-length concept with combined adaptive threshold derived by basic mean, standard deviation, and average peak-to-peak interval.

5) A binary ECG classification algorithm for accurate detection of abnormal cardiac beats. The algorithm is based on Support Vector Machine (SVM) with Radial Basis Function (RBF) kernel what makes it relatively efficient for mobile applications.

6) Two-Stage Distributed event-triggered hybrid model for assessment of health status and its deterioration monitoring. The proposed model consists of two stages:

   a) 1st stage is the distributed data filtering against medically approved ranges such as NEWS (Royal College of Physicians 2012) and events issuing when any threshold has been triggered.

   b) 2nd stage is the event-triggered data fusion algorithm based on Kohonen Self-Organising Maps (SOM) that realise the patient specific classification of health status. deterioration monitoring. Through continuous adaptation based on measures of central tendency and spread it is capable to adjust population based threshold values to a specific user therefore more precisely detect any deterioration from optimal health status.

1.5 Thesis Outline

This thesis is structured in three parts, as shown in Figure 1.5. The first part comprises the review of the literature on well-being monitoring, WSN applications as well as classification methodologies. The proposed framework for physiological signals monitoring and the analysis model are presented in part two. Part three focuses on the validation of the model and evaluation of the framework in light of project hypotheses and its primary drivers: cost-efficiency, accuracy and effectiveness of physiological signals monitoring over WSN. The thesis concludes with a discussion of the path followed in this research project and the areas for further work.

Chapter 2 gives the background information about the health and well-being monitoring, its indicators and assessment methods as well as review state-of-the-art developments in the field of WSN, tele-health and remote health monitoring.

Chapter 3 provides detailed analyses of selected vital signs, their physiological significance and accepted ranges. Approved assessment and measurement methods are contrasted with what remote monitoring can offer. These knowledge is used to analyse the testing, training and validation data sets, that were used to develop the processing model.

Chapter 4 proposes the framework for remote physiological signals monitoring on mobile devices. This include the requirements analysis, framework architecture as well as software designs for the underlying smartphone and sensor middleware.

Chapter 5 focuses on the hardware designs of the prototype system for vital signs monitoring. It includes elastic chest strap sensor node, which integrates ECG, respiratory rate and temperature monitors, as well as armband sensor node for pulse oximetry. In addition to this this chapter also presents methods and results for signal pre-processing.

Chapter 6 presents the inference engine/runtime environment and the complementary Inference Engine Editor. It contains a discussion on the development and deployment model that this editor promotes.

Chapter 7 focuses on data processing model and its building blocks, which base on mixture of statistical and machine learning algorithms. It describes how concurrent analysis of multiple physiological signals has been accomplish in order to achieve adaptive, patient-specific well-being monitoring model. Associated implementations, including testing, training and validation of the model, are presented.

Chapter 7 evaluates platform and model designs and discusses the implications of the hypothesis testing results. It highlights any limitations, the contributions and impact on the field of health informatics and paves the path for future works.

Chapter 1
An Introduction to the research

Research Problem: Cost Efficiency, accuracy and effectivenes of health monitoring
Proposed Solution: Multi-parametric, intelligent, user adaptable well being monitoring

Hypothesis 1

Development and deployment of new mobile health monitoring applications can be effective and cost efficient.

Hypothesis 2

Multi-parametric, patient specific and adaptable vital signs monitoring produce more accurate and more effective results than conventional techniques.

PART I. Towards an understanding of what well being is, how it can be measured and assessed (RQ1)

Formulation of well being concept from literature. Identification of factors affecting well being and appraisal of their significance. Review and selection of parameters that can be objectively measured. Identification of assessment methods and remote monitoring practices

Chapter 2
Ubiquitous health and well-being monitoring: Literature review

Chapter 3
Physiological signals and measurement: Data analysis

Vision: Rapid integration and simplified development process

Vision: From Illness to Wellness monitoring

PART IIa. Supportng the shift from illness to wellness monitoring thorough platform development (RQ2 & RQ3)

Determination of health monitoring requirements. Identification of functional components of wireless sensing and analysis framework and their organisation. Design and implementaton of proposed architectures, mechanisms and components.Test beds. Implementation of Inference Engine and Visual Editor. Determination of what mechanisms can be utilised and\or should be implemented in order to effectively deliver custom functionalities and the analysis logic to individual end-users.

Chapter 4
Portable vital signs monitoring framework: System design

Data Acquisition Platform

Chapter 5
Mobile Inference Engine model

Chapter 6
Prototype data acquisition platform

PART IIb. Towards a multiparamter, patient-specific and adaptable aproach to vital signs monitoring (RQ4)

Modelling of concurrent analysis of multiple physiological signals. Modelling of patient specific processing to follow patient overall health status. Determination of decision fusion and user adaptation techniques and their evaluation.

Data Processing Platform

Chapter 7
Vital Signs Classification and Decision Process

A cost-efficient multi-parametric framework for ubiquities vital signs monitoring

A new effective rapid logic-centered development and deployment model

A new accurate individualised approach to healthcare through patient-specific model for health status deterioration anlaysis

PART III. The validity and value of new aproach to health monitoring and assesment

Paltform and model evaluation. Hypotheses testing and justification. Discusion on overall results. Lessons learned on building ubiquities health monitoring systems. Differences between real-time and offline models. Relations to other monitoring and classification models. Applications.

Chapter 8
Discussion of Results and Conclusions

Cost efficiency, accuracte and effective model of healthcare that focus on wellness rather illness monitoring

Chapter 8
Summary and Further work

Figure 1.5 Outline of the thesis structure.

Chapter

2

Ubiquitous health and well-being monitoring: Literature review

2.1 Introduction

Accurate and ubiquitous health monitoring of a population by means of non-intrusive, wearable devices has been a topic of interest for the research community in recent years. Various research on this subject have been grouped together and described as m-health, a generally understood term for 'emerging mobile communications and network technologies for health care systems' (Istepanian, Laxminarayan and Pattichis 2006). It is intended to be the next generation of e-health systems, which deliver user tailored health care services with the vision of empowered health care on the move. M-health, which in principle benefits from fast and steady advances in wireless networking, medical sensors and mobile computation as well as its increasing integration in everyday life, create a wonderful opportunity to deeply modernise and change the way health care services are deployed and delivered. Following Otto's *et al.* (Otto, Milenkovic, Sanders *et al.* 2006) user tailored and ubiquitous health care approach allows us to focus on prevention, early detection and management of wellness rather than illness. Therefore, it can also be called a "pervasive health care" or health care to anyone, anytime, and anywhere, removing many crucial restraints such as time, service localisation, accessibility etc., especially for people advanced in age.

In seeking to define the best approach to achieving next generation of health care, whether it means improvements in home telemedicine, wearable ubiquitous monitoring or implantable devices, the most important factor is its context of use. Over the years, research in this topic indicated, that remote health monitoring in non-intrusive and non-invasive manner is getting more and more popular (Otto, Milenkovic, Sanders *et al.* 2006, Kirsch, Mattingley-Scott, Muszynski *et al.* 2007, Tia, Pesto, Selavo *et al.* 2008, Guo, Tay, Xu *et al.* 2009). Vital signs, which most commonly are subject to measurement, are psychological functions such as ECG, cardiac frequency, respiration, blood oxygen saturation, temperature, CO_2, O_2 and others. According to Lymberis (2003), it can also cover aspects as "sensorial, emotional and cognitive reactivity such as EMG, microcirculation, posture, fall, movements, speed, acceleration and pressure" which leads us to another important key term, namely well-being monitoring. Its simplified understanding postulates that well-being is in some way about the 'goodness' of

someone's life, what in details explain section 2.2 below. This approach creates new horizons for human conditions monitoring and can lead to the delivery of more sophisticated and more accurate results, when appropriate techniques and methods will be employed. Therefore, in answer to next-generation health care services, the crucial step seems to be the precise definition of monitoring as ubiquities. The most technologically promising approach to ubiquitous health monitoring systems utilises wireless body area networks (WBAN). As further defined by Otto et al. (2006), "WBAN consists of multiple sensor nodes, each capable of sampling, processing, and communicating one or more vital signs (…) or environmental parameters (temperature, location, humidity, light)". Another step forward might be Intelligent Biomedical Clothing (IBC), the result of multidisciplinary research in the area of mobile and wireless telecommunication, microsystems and nanotechnologies, textile & clothing, biomedical engineering, telemedicine, as well as public health and medicine. Results in this research area can lead to a transition from 'medical clothes' for patients to 'health clothes' for citizens - significant change from illness to wellness monitoring and management.

Further in this chapter, the use of non-intrusive and wearable devices for health and well-being remote monitoring is investigated, followed by motivation, research opportunities and pitfalls analysis. Finally, intended contribution of this research to the art and science of physiological signals monitoring is presented.

2.2   Well-being – more than feeling good

Literature on the nature of 'well-being' and its various meanings is massive and diverse. Initially term 'well-being', formulated centuries ago, was most commonly used in philosophy to describe what is non-instrumentally or ultimately good for a person. Long ago, Aristotle said that well-being is not merely a sensation of happiness and that human beings have more faculties than just feeling happiness, pleasure or pain. Though, popular use of this term usually relates to healthy subjects to describe how well a person's life is going for that person, both mentally and physically. However, its simplified understanding postulates that well-being is in some way about the 'goodness' of someone's life, there is no straightforward or agreed definition, which might be more extensive and deal with different aspects and various approaches. Hence, health conditions, for example, might be said to be a constituent of a person's well-being, but it can't be plausibly taken to be the only factor that imposes the status of well-being (Crisp 2001).

Health and well-being in context of various environments and settings have become common topic across different media, magazines and scholarly research publications. In seeking to define a valid health and well-being construct as noted by Kaplan *et al.* (1976) in his early studies on the health status and index of well-being, "a construct must have a valid content – that is, the content must positively and exhaustively define the dimensions of the construct and its measures". As further discusses, "since the full universe of content in the term 'health' is not yet generally agreed on, final resolution of the definitional problems can come

only through consensual validation of some proposed construct by researchers". Therefore, following this remark, focusing on the literature review in this subject in order to illustrate this term, seems to be the good starting point.

First standpoint, formed by Kaplan, Bush and Berry (1976), chose the term 'well-being' to represent the total quality of life in regard to health. They avoided incorporation of any other facets of life such as housing or income, because as they claim, it "would not reflect the relative desirability (satisfaction, utility) of housing except by virtue of implicitly assumed or consensually established levels of quality, value or well-being associated with various amounts of housing space". This assumption was chosen to preserve the validity of content and measurements by clear limitation of construct's dimensions.

Danna and Griffin, in turn, in theirs broad domain literature review, address health and well-being in the workplace from physical, emotional, physiological and mental perspectives. Their definition suggests two salient person-related concepts that are often combined with a more societal-level perspective.

> *"The first is that health and well-being can refer to the actual physical health of workers, as defined by physical symptomatology and epidemiological rates of physical illness and diseases. The second is that health and well-being can refer to the mental, psychological, or emotional aspects of workers as indicated by emotional states and epidemiological rates of mental illnesses and diseases."* *(Danna and Griffin 1999:361)*

Majeed and Brown, consecutively, enhance this model to view well-being as three separate but interrelated components: mental well-being, social well-being and physical well-being. In their well-being conceptual model, presented in figure 2.1, they suggests that well-being is determined by personal experiences of everyday life, and those, in turn, are determined by the activities we perform or we failed to perform. "In addition to this our experiences of performing or not performing activities are affected by personal and context factors, which also affect the type of activities we perform" (Majeed and Brown 2006).
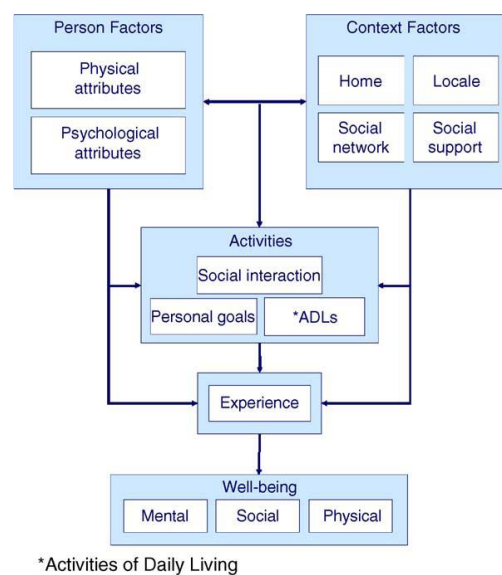
Figure 2.1 Well-being concept model (Majeed and Brown 2006:386)

Moreover, in a judgement about person's life as whole, an important role plays also so called 'subjective well-being'. This term was used by Diener (2000), who formed a number of separable components, which affect it; notably life satisfaction (global judgments of one's life), satisfaction with important domains (e.g., work satisfaction), positive affect (experiencing many

pleasant emotions and moods), and low levels of negative affect (experiencing few unpleasant emotions and moods).

Combining those views together, to synthesis what in principle constitutes personal well-being; we might in conclusion get to the diagram as the one presented in Figure 2.2. This model clearly draws from an interdisciplinary perspective of medicine, psychology and social science as well as others, because all have something to contribute to a unified understanding of well-being. Therefore, this conceptualisation, created following Maslow's hierarchy of needs example, takes well-being status as an aim in life aspirations and divides itself on four basic components, which represents different standpoints that are strongly dependent and interrelated. Similar to its exemplar, the closer to the top, the impact of this construct on overall well-being status is getting less essential. Therefore, physical health as the most bottom predictor is said to be this factor that influences well-being at most (Angner, Ray, Saag *et al.* 2009) what in other words indicates there is no well-being without health oriented wellness. Set of factors that in principle reflect physical health can be classified as 'physiological signals' or 'vital signs'. It consists of heart rate, blood pressure, body temperature as well as other parameters that provide objective judgement on someone's general health status. Such measures, taken from the human body, for instance, in the ubiquities manner, can provide the clearest answers to issues associated with health likewise well-being. Beyond their use in detection of specific physical or physiological symptoms or diagnoses related to health, physiological signals reveal strong potential for emotions recognition or stress monitoring (Jovanov, O'donnell Lords, Raskovic *et al.* 2003, Nasoz, Alvarez, Lisetti *et al.* 2004, Leon,
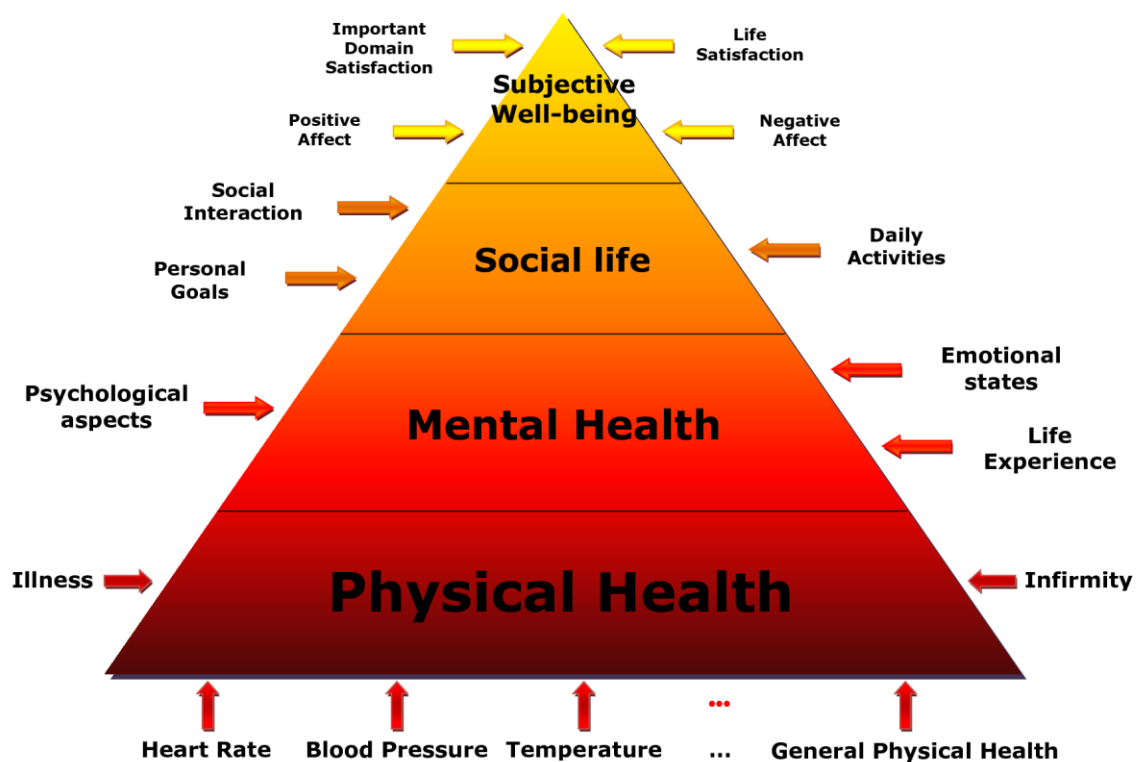
Figure 2.2 Conceptualisation of well-being framework for the purpose of future research practices, classifications and assumptions.

Clarke, Callaghan *et al.* 2007). This smooth traverse from physical health to mental health shows that physiological signals play an important role in well-being framework not only for physical but also for mental aspects.

However, physiological signals represents significant and common part of the whole set of factors that impose well-being, there are others such as those mentioned on the diagram above: infirmity, illness, life experience, emotional states and psychological aspects; but also those that go somehow behind it and constitute the upper levels of well-being hierarchy, namely social life and subjective well-being. Social life following Majeed and Brown (2006) is classified as separate level and placed above psychological health. This is determined by such facets as social interaction, daily activities and personal goals. Finally on the top of this pyramid we have grouped all other aspects, that are strongly person related and can hardly been classified as objectively measurable. We call it, following Diener's (2000), 'subjective well-being' and distinguish it on life satisfaction, important domain satisfaction as well as positive and negative affect. Nevertheless, this model serves as the most optimal summary of those voices on well-being subject analysed above, it can't be conceivable taken as the universal model. Moreover, in addition to all what have already been said about well being, there is another important aspect that has to be taken into consideration, when seeking to define exact meaning of this term across literature. Numerous empirical investigations that are conducted by various researchers, specialized in different disciplines, create plenty of definitions and measurement methods for this construct. Some researchers see well-being in horizontal, some in more vertical perspective (referring to Figure 2.2) what determine the extent to which set of different predictors are seen as a single, bigger interrelated entity. This implies terms they use to refer to well-being, which vary from 'psychological well-being', 'physical well-being', 'mental health', 'physical health', to 'subjective well-being'. In order to narrow down the well-being domain of this research, we should consider it as measurable, 'tangible' construct, described recalling above terms, as 'physiological well-being', 'physical well-being' or 'physical health'. For now such limitation should give enough accuracy.

2.2.1   Well-being and Health

The World Health Organisation (2003) defines health as 'a state of complete physical, mental, and social well-being and not merely the absence of disease, or infirmity'. In turn, the Organisation for Economic Cooperation and Development defines health as 'a physical, psychological, mental, and social state of tolerance and compensation outside the limits of which any situation is perceived by the individual as the manifestation of a morbid state' (cited in Danna and Griffin 1999). Such definitions of health encompass a broad spectrum of predictors that influence an integral perception of this term versus well-being. Hence, in order to provide some synthesis and consistency of terminology used in this research, we need to introduce some form of gradation or categorisation, for the sake of clarity.

Being consistent with previous conceptualisation of well-being and brief identification of facets that influence personal welfare perception, "health can be classified as a sub-component of well-being, which comprises the combination of such mental/physiological indicators as affect, frustration, and anxiety and such physical/physiological indicators as blood pressure, heart condition, and general physical health" (Danna and Griffin 1999). Therefore, health in such view constitutes one of many predictors of well-being, just next to emotions, life experience, social life etc. Well-being tends to be a broader concept that takes into consideration the 'whole person' (Warr 1990). Recent studies found a robust correlation between health and happiness in various populations across the life span thought the Western World (see Figure 2.3), but as noted by Angner *et al.*(2009) we have little to say about the nature of this relationship. As further noticed by Kaplan (2001), any exploration of the relationship between health and overall well-being struggle with many alternative approaches to health measuring. Through early works of Friedsam and Martin (1963), Larson (1978) or Diener and Seligman (2004) it has been revealed that subjective health measures, which are based on own rating of someone's health status are better predictors of happiness than objective measures. However, their dated and narrow set of objective measurements, use of small samples or samples that are relatively healthy, shows that this studies have some limitations and results are held back from common acceptance. Therefore, Angner et al. (2009) designed own experiment, which amongst other hypotheses should answer the question, whether subjective health measures are better predictors than objective health measures (for more information on the experiment see (Angner, Ray, Saag *et al.* 2009)). In conclusion they surprisingly found out, that while self-rated health was one of the best predictors, objective
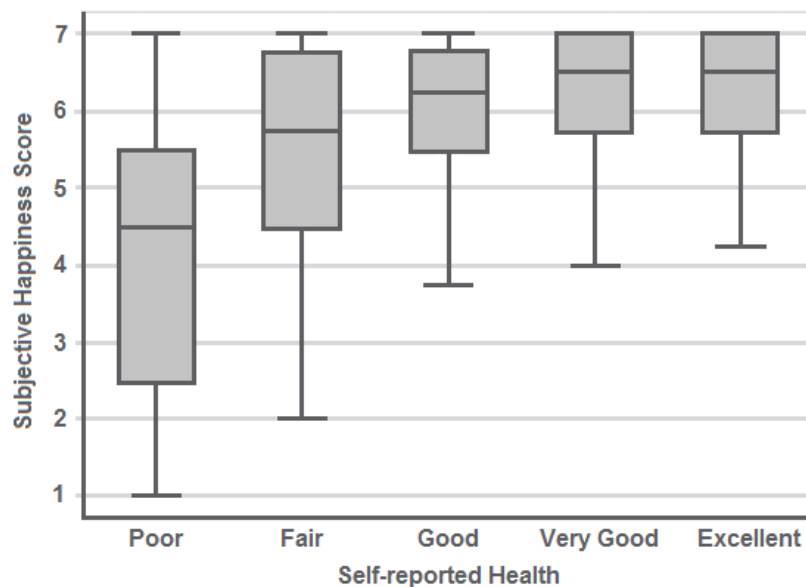
Figure 2.3 Subjective happiness scores by self-rated health (n = 382). Thick horizontal lines represent medians, boxes represent interquartile ranges and whiskers represent extreme values excluding outliers (marked by dots). Both median happiness (p < .001) and happiness variability (p < .001) differed significantly across categories of health (Angner, Ray, Saag *et al.* 2009)

measures did not predict happiness. However as further stated in their discussion on results, "insofar medical condition affects happiness at all, it will only do so for a relatively short period of time after the diagnosis or the appearance of the symptom"(Angner, Ray, Saag *et al.* 2009), because people tend to adopt to illness or try to compensate from domains others then health. This points us to the statement, that just in time detection and as tide up as possible controlling and monitoring is crucial for the whole illness prevention and wellness trend cultivation.

All above derivations head us to more 'healthy' part of our conceptualisation, which concentrates around medical facets of well-being. In respect to previous assumptions that well-being is about someone's goodness, colloquially saying, from the health point of view, if "something stops to be good" for a person, it falls into medical diagnosis framework. In dispute about meaning and use of different terms, Danna and Griffin (1999) suggest that term '"health", generally appears to encompass both physiological and psychological symptomatology within a more medical context (e.g., reported symptomatology or diagnosis of illness or disease)'; therefore, they propose to use this term, when specific physiological or psychological indicators or indexes are of interest and concern. The meaning, strength and level of medical expertise that health entails, best illustrates Figure 2.4, where three levels of expertise were used; notably measurements, monitoring and assessment. First two levels represent what in various literatures on this subject is often described as health status monitoring, that is: measuring instruments, physiological parameters such as blood pressure or pulse and their methods of measurement. However these levels encompass knowledge and methods that are relatively easy to represent in algorithm notation, things get complicated when trying to standardise the

Figure 2.4 Dimensions in classification of health and well-being monitoring

level of health assessment, which represents whole medical knowledge applied to obtained data. This is a real milestone with so many different domains, different variables, subjective

measures or subject specific values, which can lead to original and unpredictable results, that diagnosis in any single area is likely to be very risky to name, base only on limited number of predictors such as vital signs.

In conclusion, the above discussion on various definitions and concepts of well-being and its assessment gave us an overview how different researchers view the problem of well-being and how complex its assessment is. In this work, however, we are going to focus on the concept of physical well-being only and will use only such objectively measurable predictors, which does not require any or only minimal human interaction with the monitoring device. For this purpose we will use physiological signals, also known as vital signs such as: temperature, ECG, heart rate, blood pressure, respiratory rate and pulse oximetry, which are measures to assess the most basic body functions. The vital signs describe patient's overall appearance and are essential part of physiological examination, often providing important initial information that influences the direction of evaluation. They are also a highly reliable predictor of physical well-being as well as life-threatening clinical events. However, it should be noted at this point, that this research do not aim to name illness or diagnoses, but to monitor 'good state' providing some assessment of it using continuous, dichotomous or descriptive scales, as well as detect and early notify about any anomalies or emergency situations that might occur. This assumption will consequently guide this research and further discussion on measurement methods, indexes and validity of well-being constructs.

2.2.2   Well-being indicators and their validity

According to Chen *et al.* (1975), two dimensions are inherent in the social construct of health, there is "(1) level-of-well-being, the social value of an individual's level of function at a point in time, and (2) prognosis, his expected transitions to other levels, more or less favorable at future times". Figure 2.5 below ilustrate this concept at the time diagram, introducing
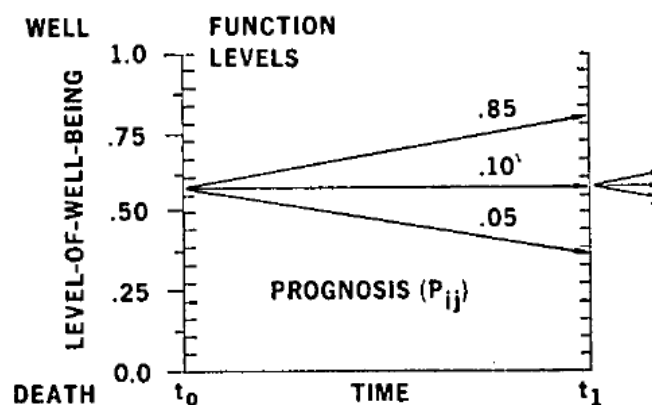
Figure 2.5 Function levels represent levels of dysfunction (disability, sensory disturbances, and symptoms) on the continuum from well-being to death. Levels-of-Well-Being (W_j) are the preferences (relative values, weights) between 0.0 and 1.0 that members of society associate with each level (j)- Prognoses (the P_{ij}'s) are the rates of transition among the function levels over time (Chen, Bush and Patrick 1975)

conceptual continous scale of well-being with a unit from 0 for deth to 1 for complete well-being. Prognoses here are the probabilities of transition between different levels or different scores, that is directed by disease or other disorders. The comprehensive index formulated by Kaplan *et al.* (1976) sees "health status as an expectation: a joint function of the levels of well-being (the weights of the states) and the expected duration of stay in each state, derived from the prognoses". However, Weighted Life Expectancy (*E*), derived in this work, represents the overall subjective well-being for population which based on expected duration of stay Y_j in each level, that cannot be directly observed, though we find that transition to the index of health status for individuals should be quite straightforward and encompass most assumptions delivered by this concept.

Therefore, comprehensive conceptual index of well-being can be expressed as

$$WB = \sum_{i=1}^{N} W_i S_i \qquad (1)$$

where: *WB*   is the index of physical well-being

W_i is the weighted impact of each vital sign *i* on overall well-being status, *i = 1… N*

S_i is the score in the range of values (0, 1) assigned to each physiological variable *i*, according to its chart of accepted ranges,

N   is the total number of physiological parameters in a given analysis

This concept could involve parameters other than physiological signs, if only we were able to find appropriate measures and weights that would reflect its impact on overall index. However, for the sake of clarity with previous assumptions, further in this work the focus is on these vital signs that we are currently able to measure (for more information see Chapter 3). Accepted ranges for these measures, mentioned above, are usually the charts of physiological signs, which classify what is meant as an optimal, normal or abnormal state. An example of such externally validated scales might be Modified Early Warning Score (MEWS) (Subbe, Kruger, Rutherford *et al.* 2001) or the American Heart Association recommendations for optimal blood pressure (Chobanian, Bakris, Black *et al.* 2003). In order to estimate correspondence between operational measurements and conceptual variables we have to look at the validity of our measure. This can be done following three basic types of validity: criterion, content and construct, however, defined by American Psychological Association (American Educational Research Association, American Psychological Association, National Council on Measurement in Education *et al.* 1999), have more general applications in variety of research.

2.2.2.1   Criterion validity

Term 'criterion validity' consider validity of the proposed measure, whether this achieves the extent that corresponds to some other observation that measures accurately the phenomenon of interest. As an example we might consider blood pressure measurement using inflated cuff and intra-arterial pressure that is measured at the same time. That how they

correspond to each other, we call 'concurrent' validity. There is also a second type of validity that forecast future criterion value and we call it 'predictive' validity.

In respect to the proposed index, the criterions for assessment of someone's physical well-being are medical measures and validated scales, which represents golden rules of physiological examination (Bickley and Szilagyi 2007). This includes methodology of physiological signals measurement using validated research instruments as well as accepted ranges for those measurements what correspond to certain medical research that classify human health conditions.

2.2.2.2 Content validity

Content validity refers to whether the items of an instrument adequately represent the domain they are supposed to measure. Items that are chosen should comply with the test specification, which is drawn up through detailed examination of the subject domain. It aims to represent that all dimensions that were considered relevant for a domain helped to accurately define it.

Thus the value of the physical well-being for our index is defined by the set of objective weighted measures that are derived from measuring devices at the given point in time. It creates time series that are sequences of data points, measured typically at successive times, spaced at often uniform time intervals. In mathematical notation time series are special case stochastic processes within time domain. In a given probability space (Ω, F, P) stochastic process is a collection:

$$\{F_t : t \in T\}$$, where each F_t is an X-valued random variable indexed by the set T - time

Bearing in mind previous assumptions and definitions of well-being, its content validity ensue from the validity of basic physiological examination, which aim to assess health status upon medical observations. The final set of parameters will be investigated upon the medical knowledge and meaning they have for overall health status and will be presented in Chapter 3.

2.2.2.3 Construct validity

Construct validity evidence entails empirical and theoretical support for the interpretation of the construct. It involves the dimensions of the construct, their domains and expected relations to each other. In other words it intends to demonstrate that an element is valid by relating it to another element that is theoretically valid and can be done through validation of theirs convergence or divergence. Convergent validity refers to the degree to which a measure is correlated with other measures that it is theoretically predicted to correlate with. Divergent (or discriminant) validity, on the other hand, describes the degree to which two measures that theoretically should not be correlated, does not in fact correlate with each other.

Proposed approach in its foundation draw upon physiological examination, which has already validated methods to assess health status (For more information on health status

assessment see (Bickley and Szilagyi 2007)). In this research we will intend to provide convergent evidence that methods of this construct relates to other measurements of the same phenomenon. Therefore, to assess validity of this construct we will correlate results obtained in this research with measurements taken using criterion methods.

2.2.3   Predictors of physical health

According to physiological examination, vital signs normally entails recording of four parameters: body temperature, heart rate, blood pressure and respiratory rate, which are the primary reference point for a patient's health status. However, this is the standard in most medical settings, the final set of parameters vary depending on context, age etc. Extended set of well-being status predictors involve: blood oxygen saturation, electrocardiograph (ECG), heart rate variability, galvanic skin response, among others. These measurements are usually supported by gender, body weight, height and age, in order to correctly classify their measurements during the interpretation process.

Vital signs are objective measures of physiological functions, which help assess the most basic body functions. However, as every human being is different, accepted ranges for each sign are recognised as a valid approximation only, which set an orientation point for further analysis. Current vital signs assessment mainly relies on comparing the measured values with medically approved normal ranges. An example of such generally accepted vital signs ranges for adults are the Modified Early Warning Scores (MEWS) used in medical admission to hospitals and as a mean of periodic monitoring of in-patients. Obtained measurements however largely depend on age, gender, race, body weight, height, and/ or individual's exercise tolerance, amongst others. Consequently, when monitoring patient's optimal health status, more complex, context-aware, and patient-specific monitoring as well as integrated analysis, have to be used in order to give more accurate results and correctly identify abnormal values and their medical significance. Hence, defining optimal set of predictors that will provide accurate results, in depth physiological signals analysis is required. For this purpose, Chapter 3 provide extensive investigation of Physiobank data (Goldberger, Amaral, Glass *et al.* 2000) and medical theory behind each parameter, in order to select the final, satisfactory set of functions that will form well-being indicators.

2.3   Remote measurement of physiological signals

Remote physiological signals monitoring systems can take different forms from implantable devices (Raatikainen, Uusimaa, Van Ginneken *et al.* 2008, Valdastri, Rossi, Menciassi *et al.* 2008), wearable sensors (Shnayder, Chen, Lornicz *et al.* 2005, Tia, Pesto, Selavo *et al.* 2008) to non-contact radar measurements (Droitcour 2006). However the most technology promising approach utilise wireless body area networks (WBAN) and wireless

personal area networks (WPAN), consisting of multiple sensor devices, each capable of sampling, processing, and communicating one or more vital signs or environmental parameters to the data processing units (Otto, Milenkovic, Sanders *et al.* 2006). WBAN includes several motion sensors, which monitors overall body activity, ECG sensors that monitor heart activity, blood oximetry devices that measure blood oxygen saturation, chest respiratory rate monitors and many others. WPAN usually consists of the local non-wearable instruments such as blood pressure cuffs, blood glucose meter, scales etc. Data processing unit or personal server, in turn, is implemented on a personal digital assistant (PDA) or personal computer (PC), and states for system control and intelligence. Since, it is core and most promising concept for non-invasive vital signs monitoring, there is a vast number of different approaches to the number of sensors, communication channels and distance as well as to the level, extend and location of the appropriate intelligence within the system.

Some research efforts such as (Van Halteren, Bults, Wac *et al.* 2004, Habetha 2006, Otto, Milenkovic, Sanders *et al.* 2006) focus more on wearable ubiquities health and well-being monitoring systems as a broader telemedical systems. This encompasses integration of different parameters, parties and advance computation techniques into one framework, where complex approach to health monitoring and just-in-time response will work towards transition from illness to wellness monitoring and management. Typical multi-tier architecture of telemedicine system is illustrated in Figure 2.6 below. According to this diagram, tier one represents body and personal area network comprised of many small, wireless and wearable sensor nodes attached to human body. The second tier is the personal server that interfaces with WBAN and WPAN sensor nodes. Interfacing to the wireless sensor nodes typically includes

Figure 2.6 Sample health monitoring network architecture (Otto, Milenkovic, Sanders *et al.* 2006)

task as: network configuration and management, selection of sensor node and theirs registration, initialisation of sampling frequency or operational mode, customisation of user specific processing procedures or set up a secured communication and data transfers. Personal server (PS) that is typically implemented on PDA or a cell phone, can manage, communicate, and analyse data obtained from sensors as well as communicate and report over WAN to the third tier which comprise of remote medical servers. This can further be passed to the medical professionals, emergency services or stored in database for further reference. Popular solutions allow also medical professionals to communicate with PS in order to forward new instructions to the user, set new trials or inspect patient's health condition. Similar models to above architecture were successfully used also by other researchers. The IBM (Kirsch, Mattingley-Scott, Muszynski *et al.* 2007), for instance, implements system, that whole interpretation logic lies on the side of medical staff; therefore it is generally design to 'shorten the distance' between patient and health professional through delivery of pure data from different remote locations to the server. This approach is very flatten and the whole system acts as a communication channel between medical staff and patients rather than an intelligent decision making tool able to significantly support treatment. On the other hand, as proposed by Augustyniak (2005), the most challenging task is to incorporate the recording node with the interpretation unit based on a PDA computer. Following his argument, this choice can be justified by easy software development possibilities and interfacing with standard peripherals such as wireless transceiver (e.g. Bluetooth, ZigBee, GSM etc.), signal acquisition module and extensible memory buffer by means of flash memory cards. The surveillance network proposed in this paper, based on the real-time adaptation model, which provides a highly flexible vital signs monitor, through combination of different levels of modification (hardware, software and interpretation parameters), "opens the possibility of deep changes of device functionality and functions" (Augustyniak 2005). However, based on current advances in mobile computing, the aspect of adaptation, which in this case was provided at the level of specialty designed device, can be easily transferred to a general purpose mobile device, such as PDA or smart phone, which is capable of managing the sensor nodes on the wireless body area network as further proposed by (Wu, Bui, Batalin *et al.* 2008).

Wireless medical telemetry is not a new concept. Growing interest in this area of bioengineering was induced by the ability to augment medical telemetry with tiny, wearable, wireless sensors as well as by the fast and steady advance in wireless communication, mobile computation, energy efficiency and miniaturisation. Therefore, there is already a number of remote health monitors available on the market, including electrocardiographs (ECGs) (Lifesync Corporation 2009, Healthfrontier Inc n.d., Alive Technologies Ltd. n.d.), pulse oximeters (Welch Allyn Inc 2009, Alive Technologies Ltd. n.d.), blood pressure monitors (Lifewatch Ag n.d., A&D Medical Inc n.d.), heart rate monitors (Zephyr Technology 2008, Alive Technologies Ltd. n.d.), respiratory rate monitors (Ambu Sleepmate Inc. n.d.) etc. However, their limited range of services, operation as stand-alone systems, lack of flexibility in integration with other devices,

data processing that happens offline and finally theirs cost have held them back from popularity and wider use. Therefore, in section 2.5 in further discussion and overview of current and already finished projects an attempt has been made to identify those factors in the system's architectures that might lead to more accurate, user tailored and intelligent processing of information at an affordable cost.

2.4   Motivation for health and well-being monitoring

Analysing current trend in population growth, current health care systems and technology advances, we can expect that wearable systems for continuous health monitoring are a key technology helping in the transition to more proactive and affordable health care. According to Smith et al. (2006) "current obstacles to this approach are that there is no commercially available device that monitors the full range of necessary variables". Therefore, as early as possible detection and management of wellness rather than illness approach, proposed in this research, have strong demographical and economical rational, vast number of benefits and equally many utility and applications in health care. Hence, the following rationale and motivation for this system are identified.

2.4.1   Demographical and economic considerations

Ubiquitous health care approach, which allows us to concentrate on prevention, early detection and management of wellness, is particularly important nowadays, when the populations of well-developed countries are going to be influenced by two important factors; on the one hand, an aging population due to increased life expectancy, and on the other hand, the Baby Boomers' demographical peak. According to the U.S. Bureau of the Census (U.S. Census Bureau 2000), the US population aged 65 or over is expected to nearly double from 35 million to 70 million by 2025, when the youngest Baby Boomers retire. Supporting it with findings of the United Nations report (United Nations Population Division 2007), we can observe that this trend is global and, for instance, the more developed regions' population over 60 is expected to increase from 245 million in 2005 to 406 million in 2050, while at the same time its total population will remain largely unchanged at 1.2 billion. All of these factors will have a significant impact on high-rising costs of healthcare liabilities, with only U.S. expenditures rising to the level of $4.4 trillion—more than double 2007 spending and exceeding 20% of the Gross Domestic Product (GDP) in the next 10 years (The National Coalition on Health Care 2009). Raatikainen et al. (2008) in theirs report on the first experiences with the implantable cardioverter defibrillator (ICD) monitoring system in Europe, as a secondary objective assessed the economic impact of remote ICD monitoring. The system, besides of previously implanted ICD devices, consists of a portable patient monitor, a central database server and a website, where clinicians can analyse delivered data. Forty-one patients with ICD devices were selected and undergone 9 month trial,

whereas theirs in-office visits were substituted by remote data transmissions. As a result, remote monitoring diminished the direct and indirect annual costs of ICD follow-up by €524 what equals to 41%. Compared, the total costs of generally applied follow-up among 41 participants accounted for up to €52k, whereas €30k respectively, following study protocol. However these results cannot be directly extrapolated on such area as physiological signals monitoring, because of the complexity of the cost calculation and theirs dependence upon the situation, nevertheless, it is likely that remote monitoring would save a substantial amount of time and money. This underscores the need for a more accurate, pre-hospital and prevention oriented health care system, which will take care of a person's physical health conditions at their earliest stage, through physical activity management, status monitoring, assessment as well as early notification in case of an emergency situation.

2.4.2   Benefits of ubiquitous well-being monitoring

First and foremost important benefit of wearable sensors is that individuals can closely monitor changes in their own vital signs and maintain an optimal health status independently from the context of use and environment. Acquisition of many different parameters and continuous adaptation of the interpretation depth allows us to follow the patient's state and diagnosis goals in various situations. Providing highly adaptable systems, by designing intelligent algorithms, capable of self-learning from a set of training data, such as historical data describing a particular relationship or expert opinions, would help in keeping patients physically active, providing early notification and detection of potential illness risks and finally more accurate treatment (Hass and Burnham 2008). Moreover, by integrating this system into wider telemedical framework, it can alert to the medical personnel or emergency services when life-threatening changes occur. This just-in-time and immediate service would be highly desirable for patients with cardiovascular problems, epilepsy as well as elderly who require continuous help from social carers. This might help those people remain living more independently in their homes, work and do all normal live activities, without worrying about theirs life. Finally, all those healthy aspects lead to the economic benefits in reduction of healthcare expenditures and insurance liabilities.

2.4.3   Utility of vital signs measurements

Vital sign, which are the primary reference point for the most basic body functions, are still rarely stored anywhere other than on a paper chart, in spite of being a valuable source of different levels of information. This data recorded over time as well as theirs noted trends, can provide valuable diagnostic tools. First, the records can be used to verify whether patient is alive and his life is not threatened. Moreover, continues monitoring, for instance prior to hospital admission, can supply valuable information on current patient's health conditions as well as

provides an insight on the patient's history in order to help to identify any abnormalities. The other possible medical applications are: diagnostic procedures, emergency response at the disaster scene and ambulatory settings, optimal maintenance of chronic conditions, monitoring the adherence to treatment guidelines or the effects of drug therapy. Apart of this, continuous vital signs monitoring is a valuable method in sport - to assess athletes performance, in military - to monitor soldiers in action, at the disaster scene - to monitor selected parameters during work in fire-fighter's' clothing (Zephyr Technology 2008) or in the space exploration – to monitor the health of astronauts to ensure their safety during space mission (Nasa n.d.). Another interesting utility of vital signs measurements is the emotions recognition and studies on theirs affect and cognition on multimodal human computer interaction (HCI) (Lisetti and Nasoz 2004). All of these potential applications give important remarks on the spectrum of physiological signals utility, what is worth taking into consideration, when modelling the successful system.

2.5   Related works

A variety of wireless personal vital signs monitors, both for medical and fitness purposes, are either already on the market, or under development at prototype stage. Competitive list of examples of wearable systems developed and published in recent years along with a brief description of their applications was presented by Chan *et al.* (2012). Some selected projects of similar nature to one proposed in this work can be found in Table 2.1 below. A vast number of those projects focused at on-body sensing technologies through integration of micro-nano technologies and flexible systems in textile material. They aimed at the implementation of the "e-textile" paradigm, where sensing, actuating, communicating, processing and power sourcing are seamlessly integrated on a textile.

Whereas sensors and actuators are essential to promote SWS adoption amongst the population, they are only means of data collection. The true benefits of health monitoring systems come with data processing and integration. These early systems, were often the side effect of sensor development, generally designed to 'cut the cord' between the patient and the medical professionals, providing mainly only instantaneous single-parameter assessment and transmission. Hence, in order to fully explore the benefits offered by SWS, current research efforts in this area focus on integration and interoperability aspects as well as new classification algorithms which will further boost SWS's adoption and release their commercial value.

2.5.1   Physiological signals monitoring

Some recent works on m-health systems has already taken the approach of providing widely understandable accurate monitoring of a population by means of non-invasive and quasi-wearable devices, for the purpose of monitoring patients during large-scale disasters (Tia, Pesto, Selavo *et al.* 2008), sport performance monitoring (Zephyr Technology 2008),

Author	System description	Applications
Curone *et al.* (2010)	ProeTEX smart garment	Health-state parameters, environmental variables
Otto *et al.* (2006)	Ubiquitous Health Monitoring	Wireless Body Area Sensor Network for Ubiquitous Health Monitoring
Coyle et al. (2010)	Biotex	BIO-Sensing TEXtiles to support health management
Katsis et al. (2011)	Aubade sensor system	A wearable system for the affective monitoring of car racing drivers during simulated conditions using EMG, ECG, respiration, skin conductivity (EDR)
Jovanov et al. (2003)	Wireless intelligent sensor system	Heart rate variability for stress measuring
Rimet et al. (2007)	Bootee	Wearable multi-parametric monitor
Anliker et al. (2004)	AMON	High-risk cardiac/respiratory patients
Wu et al. (2008)	MEDIC	Medical Embedded Device for Individualized Care that monitors: pulse, temperature, and heart rate
Haahr et al. (2008)	Electronic patch	EMG, arterial oxygen saturation
Ma et al. (2011)	Electronic second skin	Temperature, ECG, EMG,
Sung et al. (2005)	LiveNet mobile platform	Accelerometer, ECG, EMG, galvanic skin conductance
Di Rienzo et al. (2005)	MagIC vest	Textile-based wearable device for biological signal monitoring
Knight et al. (2005)	SensVest	Vital signs: movement, energy expenditure, heart rate, body temperature
Pandian et al. (2008)	Smart Vest	Wearable multi-parameter remote physiological monitoring system measuring: ECG, PPG, heart rate, systolic and diastolic blood pressure
Shnayder et al. (2005)	CodeBlue	Mote based system for pulse oximetry, ECG, EMG and activity
Zephyr (2012)	BioHarnesses	Commercial remote physiological monitor measuring heart rate, breathing rate, posture, activity level and peak acceleration
Andre et al. (2006)	BodyMedia	Commercial energy assessment armband device to assess physical activity and lifestyle
Hexoskin (2012)	Hexoskin	Commercial wearable body metrics for athlete's performance monitoring
Grossman (2004)	LifeShirt	Commercial multi-function ambulatory system monitoring health, disease, and medical intervention in the real world
DeVaul et al. (2003)	Mithrill	Research platform for context aware wearable computing
Habetha (2006)	MyHeart	Cardio-vascular diseases telemonitoring platform for preventive lifestyle & early diagnosis
Valdastri et al. (2008)	Implantable telemetry platform	Gastro oesophagus pressure, pH, glucose monitoring

Table 2.1 Personal Health Monitoring systems overview.

chronically ill patients in their home environments (Kirsch, Mattingley-Scott, Muszynski *et al.* 2007), diabetes monitoring (Lee, Lee, Ha *et al.* 2009), individualized health monitoring and diagnosis (Wu, Bui, Batalin *et al.* 2008) or to monitor astronauts life functions during space travel (Mundt, Montgomery, Udoh *et al.* 2005). Moreover, some commercial, off-the-shelf devices, have appeared on wellness and lifestyle market, such as Samsung miCoach (Samsung Ltd 2008) - mobile phone coaching system for runners or Polar hearth rate monitors (Polar Ltd n.d.), both measuring heart rate, but providing no further information about ECG, respiratory rate etc. Some other devices such as Body Media (Body Media Inc. 2009), are additionally capable to measure more parameters such as body activity, temperature, and galvanic skin response.However, due to high cost, limited range of services, lack or offline data processing, unwieldy wires between the sensors and the monitoring systems, lack of flexibility and integration with third-party devices or inappropriate hardware and software solutions, make such systems impractical for continual monitoring and early detection of medical disorders, what cause that wider acceptance of such systems is still limited (Lymberis 2003, Augustyniak 2005, Otto, Milenkovic, Sanders *et al.* 2006, Chan, Estève, Fourniols *et al.* 2012).

Reviewing ongoing and finished projects, p we can notice that there are several examples of research projects aiming to develop health assisting devices using different methods and since some of them such as already mentioned IBM or miTag projects, are generally designed to 'cut the cord' between patient and medical professional, others such as AMON (Anliker, Ward, Lukowicz *et al.* 2004) takes a step further providing real-time data analysing as well as integration and optimization of sensor devices, for instance, to fit all-in-one wrist-worn system for high-risk cardiac/respiratory patients. This system, as most of them does, continuously collects and evaluates multiple vital signs (in this particular example blood pressure, SpO_2 and one lead ECG); however by using an unobtrusive wrist-worn device, it is performed without interfering with the patients' everyday activities and not restricting his/her mobility.

First, fully commercialized project constituted of WBAN devices, which shapes the future of integrated mobile health monitoring is MobiHealth System (Mobihealth B.V. 2009) released by Mobihealth B.V., company founded in 2007 after successful research conducted under the scientific lead of the University of Twente, Netherlands. It provides an integrated mobile remote monitoring and feedback system that integrates with compact third-party sensor systems through Bluetooth industry standard interfaces. Supported physiological monitoring functions include: multi-lead ECG, multi-channel EMG, plethysmogram, pulse rate, oxygen saturation ($SpO2$), respiration, core/skin temperature. In spite of having intelligent capabilities to analyse acquired data, the main aim of this project is to ensure that patients stay securely connected to their remote care professional. Other medical oriented commercialised products such as MedApps (Medapps Ltd. n.d.) or Polymap (Polymap Wirelles 2008) either use specially designed devices to collect and analyse data, what increase the cost of such system development or aim towards WPAN instead of WBAN implementation.

Among the research projects, that take step forward in data analysing, is MEDIC (Wu, Bui, Batalin *et al.* 2008) which states for Medical Embedded Device for Individualized Care and is a PDA-based generic architecture for wearable systems, which can be used in patient monitoring and medical diagnosis. The biggest advantage of this system over other solutions is a diagnostic engine for local signal processing and detection of patient conditions, as well as for making decisions in managing the system's sensors. As the approach to health care shifts from being 'hospital-centred' to 'patient-centred' (Lymberis 2003), this system, taking advantage of improvements in computation speed, memory capacity and GUI made to current mobile devices, adopt to this trend, by changing from passive patient monitoring device with fixed functionality to a proactive agent that may interact with a patient and self-reconfigure to provide the most certain inference of a patient sate. Implemented system for gait analysis provides real-time monitoring and classification of patient state, using Fast Fourier Transform (FFT) for signals feature extraction and Naive Bayes Classifier to infer the probabilities that the patient is in one of health classes. In results system is able to detect whether patient is limping, using two 3-axial accelerometers, whether the limp is on the left or right leg and to determine the severity level of the limp, using additional knee sensors, requested when limp is detected. This work presents efficient, ad-hoc use of resources when required, network reconfiguration capabilities as well as possibility and advantages of using an artificial intelligence for real-time medical diagnosis and decisions making.

Since most of these projects utilise any portable device as personal server, they have to implement some software architectures, which can communicate, manage and send information to the sensors or further to the medical service providers. In the design of such a general-purpose, relaxed model, the framework proposed by Otto et al. (2006) can be very useful in case of system organisation and wireless devices interconnection. It provides a complex understanding of this system and in detail concentrates on software architecture including connections scheduling, devices calibration with use of beacons as timestamps, events handling, data capture and management. Current attempt to provide reusable mobile-centric wireless sensing platform presents NORS (NOkia Remote Sensing) (Trossen and Pavel 2005). According to the Remote Sensing Architecture, N-RSA is an event-based architecture that allows for collecting and aggregating sensor data obtained locally at mobile device gateways. The underlying event subscription model of the N-RSA enables to transfer only relevant data. With this, the existing intelligence in the mobile devices is exploited to reduce data traffic and distribute system logic. Another interesting approach to remote sensing presents MobHealth (Mobhealth n.d.), an extensible, open source, mobile health framework/API written in JavaME. The current version supports two sensor-data formats (e.g. used by AliveTec (Alive Technologies Ltd. n.d.)) and already provides higher-level classes for data recognition like body position or hypoxia-detection. Transparent remote framework for wireless physiological signs sensing with customisable interfaces, customisable data transfer protocols or 'plug-and-play' capabilities to easily interconnect devices, might lead to rapid health application development,

improvements in systems optimisation and cost reduction; however design constraints and preliminary state of current projects hold them back from popularity.

2.5.2 European perspective

In Work Programme 2014-15 for the Horizon 2020 (European Commission 2014), the European Commission highlighted "Personalising health and care" (see Work Programme part 8 – Health, demographic change and wellbeing), as one of the Europe's key socio-economic challenges. It focuses on ICT for disease prediction, early diagnosis, prevention, minimally invasive treatment, overall disease management and support to healthy lifestyles. As part of the target outcomes for this challenge the Intelligent Personal Health Systems (PHS) for remote management of diseases, treatment, rehabilitation or analysis of multi-parametric data, amongst others, are expected. However such systems could also apply to different domains, where a person's physical condition is an important factor, such as sport, computer games, military, automotive etc.

One of the biggest European initiatives into health and wellbeing monitoring founded by EPSRC in UK, is project SPHERE (Craddock 2015) started in December 2013, which aims to develop sensor technology specifically for individual disease conditions monitoring. Project vision is to impact a range of healthcare needs simultaneously by employing data-fusion and pattern-recognition from a common platform of largely non-medical/environmental networked sensors in a home environment. Although this project is still in its early phase it is a very interesting and promising approach, which employs a similar approach to the one proposed in this work, but taken to a large scale integrative deployments for testing validation and refinement. However the main difference between this project and project SPHERE is its Work Package 5: Data fusion and data mining, which from early public releases seem to deploy data processing logic on central medical servers what is the opposed approach to the one taken in this research project.

Further, numerous EU projects focus on monitoring of vital signs of elderly patients, mostly at home. This include projects such as OLDES (Older People's e-Service at Home) (Busuoli, Gallelli, Haluzík *et al.* 2007), CAALYX (Complete Ambient Assisting Living Experiment) (Kamel Boulos, Lou, Anastasiou *et al.* 2009), K4CARE (Knowledge-based Home Care eService for an Aging Europe) (Campana, Moreno, Riaño *et al.* 2008), ENABLE (a wearable system supporting services to enable elderly people to live well, independently, and at ease) (Parker, Nussbaum, Sonntag *et al.* 2008), SOPRANO (Service-Oriented Programmable Smart Environments for Older Europeans) (Sixsmith, Meuller, Lull *et al.* 2009), REACTION (Remote Accessibility to Diabetes Management and Therapy in Operational Healthcare Networks) (Ahlsén, Asanin, Kool *et al.* 2012), INHOME (an intelligent interactive services environment for assisted living at home) (Vergados, Kavvadias, Bigalke *et al.* 2008), MonAMI (Mainstreaming on Ambient Intelligence) (Ibarz, Falcó, Vaquerizo *et al.* 2012) and Hydra (networked embedded

system middleware for heterogeneous physical devices in a distributed architecture) (Eisenhauer, Rosengren and Antolin 2009). Additionally, PERSONA (Perceptive Spaces Promoting Independent Aging) (Soler, Peñalver, Zuffanelli *et al.* 2010) and UniversAAL (ambient assisted living (AAL)) (Tazari 2010) provide solutions for user activity monitoring. These platforms transmit data on the health status, risk assessments and self-management incitements from sensors and devices at the patient's home to health-care professionals, informal carers and emergency and crisis management teams. All these projects unlike the proposed system monitor only a predefined, limited number of vital signs parameters from preselected sensor devices and/or focus on elderly patients only. In turn, the proposed system's main focus is at well-being management and optimal health status maintenance of any individual. Moreover proposed model aims to limit the communication between patient and health professionals through intelligent, real-time data processing, which only transmits health status or meaningful events when these is either requested by the healthcare professionals or third party caregivers or in emergency situation.

2.5.3   Sensor devices

Over the last few years there has been a significant increase in the number and variety of wearable health monitoring devices, ranging from simple activity monitors, pulse monitors and portable ECG monitors, to sophisticated and expensive implantable sensors. Some of commercialised products such as Sensatex, LifeShirt, and MagIC, embed sensors inside clothes (Grossman 2004). Others such as Actis (Otto, Milenkovic, Sanders *et al.* 2006) at the University of Alabama, BodyNets (Dabiri, Noshadi, Hagopian *et al.* 2007) at UCLA or CodeBlue at Harvard University (Shnayder, Chen, Lornicz *et al.* 2005), detect vital signs using wireless body nodes design such as Mica2, MicaZ or TelosB (Crossbow Technology Inc. n.d.), equipped with smart sensors. It was possible, thanks to continues significant advances in minimisation and development of new wearable sensors, what today leads us to such new devices as wearable blood pressure sensor that offers 24/7 continuous monitoring (Trafton 2009) or textile sweat sensor to measure sweat pH and sweat-rate during exercise (Coyle, Morris, Lau *et al.* 2009). Moreover, some improvements were made to measurement methods such as continuous non-invasive blood pressure measurement by pulse transit time (Fung, Dumont, Ries *et al.* 2004), utilised by (Tay, Guo, Xu *et al.* 2009) or Doppler radar measurements of heart rate and respiration (Droitcour 2006). Finally, such rapid increase in the number and variety of wearable health monitoring devices wouldn't be possible without introduction of embedded operating systems for sensor networks. The most popular one, TinyOS (Tinyos n.d.) is a lightweight open source component-based operating system written in the nesC programming language. NesC originates from C programming language and was optimized for the memory limitations, task synchronization and task management. TinyOS applications are a set of

component modules, what results in natural modular design, minimal use of resources and short development cycles (Otto, Milenkovic, Sanders *et al.* 2006).

By using commercially available, Health & Safety certified, non-invasive off-the-shelf wearable devices, we are able to measure, among the others, the following vital signs:

- Blood pressure (Systolic/Diastolic) - using elastic wireless arm cuff with automatic inflation and deflation, such as the one manufactured by Mytech Technology Ltd (n.d.)

- Heart rate and blood oxygen saturation (SpO_2) - using plastic housing, that slips over the index finger or earlobe and contains an array of LEDs along one inner surface and an optoelectronic sensor opposite. Pulse oximetry involves the projection of infrared and near infrared light through blood vessels near the skin. By detecting the amount of light absorbed by haemoglobin in the blood at two different wavelengths (typically 650 nm and 805nm), the radial pulse signal can be measured and further used to calculate heart rate and oxygen saturation. Device with built-in wireless transmitter that presents such capabilities is Alive Wireless Pulse Oximeter from Alive Technologies Ltd (n.d.)

- Respiratory rate - using chest plastic strip with piezoelectric sensor to measure chest and abdominal expansion associated with respiratory effort, from Ambu Sleepmate (n.d.)

- Temperature - using wireless temperature sensor based on thermo transistors, such as the one developed by Gentag (n.d.)

- ECG – using electrodes attached to upper and lower chest to measure cardiac activity, through short sampling of the heart's electrical activity between different pairs of electrodes such as Alive Heart Monitor developed and manufactured by Alive Technologies Ltd. (n.d.)

Supported by (Lymberis and Olsson 2003, Istepanian, Laxminarayan and Pattichis 2006, Patel, Park, Bonato *et al.* 2012) and others, there is a strong believe that further improvements in bioengineering microsystems and nanotechnologies, textile & clothing, will lead us to new sensor technologies in the near future. Bearing in mind further reduction of size and improvements in measurements accuracy, the focus of this project is on health/well-being data sampling, management, and processing using mobile technologies, taking current state-of-the art in sensors technology as it is, with strong hopes on improvements in usability and design of new compact, sensor devices in the future.

2.5.4   Communication technology

According to Valdastri *et al.* (2008) "connections with leads and cables present obvious disadvantages: they limit patient's mobility and, moreover, they can cause skin irritations or infections, thus contributing to deteriorate health conditions". This is the main motivation for modern system to use wireless technology, when interconnecting sensor devices. It allows an integration of various nodes into WBAN. Current standards for wireless communication include WLAN, Bluetooth and ZigBee (Gungor and Hancke 2009). WLAN derives from IEEE 802.11

while other two solutions originate from IEEE 802.15 Standard family and all use the 2.4 GHz ISM frequency band.

WLAN, commonly known as Wi-Fi or wireless LAN, is intended for resident equipment and replacement for cabling for general local area network access. Despite the radio frequencies which are the same as Bluetooth and ZigBee, it was developed for very high data rates (tens of Mbps) and characterise with higher power consumption, therefore resulting in a stronger connection over longer distance. This however discard common use of Wi-Fi in health monitoring, since biomedical devices needs to be as small as possible, energy efficient and while at the same time without a need for high data throughput. Nevertheless, incorporation of WLAN in the bigger telemedicine system is still worth consideration even to communicate over WAN with remote sites such as medical servers or practitioners (Otto, Milenkovic, Sanders *et al.* 2006).

Bluetooth, as described in system specification, is the short-range wireless technology intended to replace the cable(s) connecting portable and/or fixed electronic devices, with focus on portability. Bluetooth has already become standard in various application including PDA, mobile phones, headsets etc. what makes it easy accessible and popular. For biomedical applications like these it also match requirements such as low battery consumption, relatively high data rates (with EDR in version 2.0 up to 3,1 Mbps), easy ad-hoc interconnecting into piconets and small size of device, what was discussed along many previous WSN applications by (Wu, Bui, Batalin *et al.* 2008, Tay, Guo, Xu *et al.* 2009).

ZigBee, is based on the IEEE 802.15.4 protocol and used by companies such as Philips, Motorola, Atmel, and Mitsubishi. It differs from Bluetooth by operating at lower rate data transmission (250kbps), ultra-low power consumption (thanks to its ultra-low power sleep mode and very short wake up time) as well as possibility to create very complex networks with up to 65,536 devices. This technology offers promising opportunity for the development not only wearable but also implantable networks as well as mix of such approaches incorporated in one system. For instance, ZigBee capable sensors can be implemented inside swallow able capsules, which can transmit signal travelling through gastro-intestinal tract, with no need for battery replacement for several years. ZigBee successful implementation for biomedical application was presented by (Tia, Pesto, Selavo *et al.* 2008, Lee, Lee, Ha *et al.* 2009).

The main criteria in the wireless technology selection process, which directly corresponds to cost-efficiency requirement, is its compatibility across various handheld devices available on the market. The selection of Bluetooth as a suitable technology for the this prototype application was based on the intention to maximise the infrastructure reuse. Bluetooth wins this competition as it already powers millions of health and medical devices, with tens of millions more on the way. Although Bluetooth is the most popular wireless technology for personal medical devices, it is not free of some issues in terms of security and data reliability when it comes to older versions of Bluetooth (v.2.1, v2.0 or earlier) still present in mobile phones today, as discussed by (Mare and Kotz 2010).

Metric	Bluetooth	ZigBee
Power Consumption	Days	Years
Number of nodes	Up to 7	Up to 65,536
Typical network join time	3 seconds	30 milliseconds
Out-of-box compatibility with mass-market devices	Yes	No
Data Rate	1 Mb/s	250 kb/S
Stack Size	100+ kbyte	8-60 kbyte
Modulation technique	Frequency Hopping Spread Spectrum (FHSS)	Direct Sequence Spread Spectrum (DSSS)
Range	1 or 100m, depending on radio class	Up to 70m
Topology	Star	Star, cluster, mesh
Security	64 bit, 128 bit	128 bit AES

Table 2.2 Bluetooth and Zigbee wireless technologies features comparison.

The security concerns include aspects of secure pairing, used to secure communication between devices as well as privacy concerned by disclosing information such as device name, address, local clock and other characteristics needed to connect to it. This is especially important for health monitoring sensors of a compact form and links with 'Just Works' (JW) pairing models designed to support devices that do not have any input-output capabilities. Unfortunately, with JW model an attacker can falsify the input-output capabilities of his device and use this association model to launch man in-the-middle attacks. (Haataja and Toivanen 2010). The privacy concern refers to 48-bit unique Bluetooth Device Address (BD ADDR) used when device operates in discoverable mode. It can be used to identify the manufacturer and hence may be used to identify the type of sensor. Moreover it can be used to link all sensor data back to one device and maybe to one user. It can also be used for tracking location of the device, if the device's BD ADDR is observed by several Bluetooth base stations in different locations.

Data reliability concerns, as authors in (Mare and Kotz 2010) suggests, include aspects such as interference, scalability and power consumption. Although Bluetooth works well against interference from sources such as Wi-Fi and Zigbee, the Adaptive Frequency Hopping (AFH) does not make it resistant to interference from other Bluetooth piconets, what can be a problem in crowded places, such as hospitals and subways, and eventually affecting the quality of service of a BAN. The other concern is Bluetooth scalability, as the device can talk to only seven other devices at a time and so this limits the number of sensors a person's BAN can use concurrently. Finally by looking at Table 2.1 we can notice that power consumption is another concern for existing Bluetooth devices when compared, for example, to ZigBee devices, which can last on one battery for several years.

What is promising about Bluetooth is its recently released more secure, reliable and more energy efficient protocol modification (Gomez, Oller and Paradells 2012), called Bluetooth

Smart (also called Bluetooth Low Energy) which address all concerns highlighted by Mare and Kotz (Mare and Kotz 2010). This protocol which is compatible with existing Bluetooth devices, offers a new wireless standard for medical and wellness applications, whether in a hospital environment or at home. Just like Bluetooth, Bluetooth Smart operates in the 2.4 GHz ISM band. Unlike classic Bluetooth, however, it remains in sleep mode constantly except for when a connection is initiated. The actual connection times are only a few mS, unlike Bluetooth which would take ~100mS. The reason the connections are so short, is that the data rates are so high at 1 Mb/s. Bluetooth Smart is also more security and reliability, through 128-bit AES encryption, which ensures a highly secure network, keeping patient and medical information confidential during transmission. The frequency-hopping radio virtually eliminates network interference and provides strong immunity to RF noise sources such as electrosurgical devices and common household appliances. With a transmission range up to 100 meters (300 feet), Bluetooth Smart delivers power efficient, wireless connection flexibility.

Finally, it is important to highlight that framework proposed in this work, does not limit to one wireless technology, but in turn enables multiple wireless technologies to operate under one framework. This is enabled by the network, nodes and sensors abstraction and virtualisation, which are discussed in the reminder of this work. Therefore Bluetooth technology used in this work should be considered as a cost-effective proof-of-concept and one of many wireless technologies that can be incorporated into the proposed framework.

2.6 Potential contributions to knowledge of physiological signals monitoring

2.6.1 Areas with potential for improvements

In undertaking this literature review, the aim was to analyse health monitoring subject in order to identify where the significant gaps lie. In summary, the most significant areas of mobile computing on the healthcare service, with need of further work are:

- Interpretation and adaptation paradigms in the context of human health and well-being.
- Studies on more sophisticated use of artificial intelligence methods on PDA/mobile phones devices.
- Design of appropriate intelligent algorithms and interpretation logic, leading to an extension of the range of services provided by commercial portable devices.
- Studies on portability of the system and what follows efficiency and low-cost of computation.
- Studies on real-time intelligent decisions making, to move towards solutions which will be sensitive to personalized data and its decisions will be tailored to the user's actual health conditions.

- Derivation of well-being descriptive algorithms based on vital signs intelligent data analysis.
- Improvement of event trigger and auto alerting algorithms in case of abnormalities detection, in order to minimize the percentage of false alarm and reduce the need for offline computation.
- Development of new sensor technologies and incorporation of new devices into integrated health monitoring systems.
- Studies on 'textronics' - electrical and electronic textiles and clothing, what is a synergic connection of textile industry, electronics and computer science with elements of automatics and metrology knowledge in producing and designing textile material integrated with electronic circuits.
- Shift to knowledge based expert systems for automated medical diagnosis.

2.6.2 Intended novelties of this project

The main aim of this research project is to model and build an integrative prototype of a non-intrusive, portable integrated intelligent system to monitor and make real-time decisions based on the assessment of healthy subjects' vital signs in the context of human's well-being conditions. The system intends to be adaptable to user, becoming, over use, a user tailored device.

In order to achieve this aim it focuses on accurate and ubiquitous monitoring of vital signs, by means of non-intrusive, wearable sensor devices deployed in WBAN. To make real-time intelligent decisions based on collected data, feasibility of using artificial intelligence methods on PDA/mobile devices is investigated. Further, on such a ground, we develop the intelligent mobile computation techniques for interpretation of vital signs and adaptation of monitoring paradigm to the current health and well-being status. Therefore, portability of the system and, what follows, efficiency and low-cost of computation, intelligence and adaptability of the algorithms as well as the data pre-processing activities have a crucial role in this project. This approach will allow focusing on prevention, early detection and management of human wellness without interfering with the patient's everyday activities and not restricting his/her mobility.

The main novelty of the system is an intelligent conceptual well-being model, based on assessment of physiological signs against approved signal ranges. Sample set of sentinel values that are used in this project, are presented for example by NEWS (generic) or Blood Pressure (signal specific) charts.

Complex implementation of intelligent well-being model (algorithms) and theirs integration in the monitoring system on mobile device, form the second developmental contribution of this project. In order to satisfy this requirement, the pivotal part of the system is the integrated inference engine/AI runtime environment that allows the mobile device to serve as an intelligent,

user-adaptable personal health assistant (PHA). The development of a complementary visual Inference Engine Editor, allows the AI specialist along with medical experts to build data processing models by gluing together different components and controlling the application logic with scripts. The editor enables to instantly deploy such models remotely on patient mobile device. This approach should shift the focus from software and hardware development, to medical and health process implementation what will speed up development and deployment of medical and health decision processes. All this aims to directly impact on response time and accuracy of medical interventions.

Third innovation is a development of transparent, remote and mobile sensing framework, based on J2ME technology for wireless physiological signal sensing. It should satisfy following additional requirements to those presented in NORS project (Trossen and Pavel 2005)

- Incorporation of an artificial neural network engine e.g. Java Object Oriented Neural Engine (Joone 2007), to allow computation and analyses of vital signs in real-time.
- 'Plug-and-play' capability to easily interconnect devices
- Easy customisable interfaces for remote sensing
- Customisable data transfer protocols between platform and sensors

Finally, in order to increase accuracy and complexity of well-being monitoring, this project aims to integrate different physiological signals monitors into one framework, therefore widening set of monitored variables by one system, however at the same time trying to optimise number of separate sensor devices.

Chapter

3

Physiological signals and measurement: Data analysis

3.1 Introduction

This chapter provides a detailed analyses of selected vital signs, their physiological significance and accepted ranges. Approved assessment and measurement methods are contrasted with what remote monitoring can offer. This knowledge is used to analyse the testing, training and validation data sets, that were used to develop the processing model.

3.2 Physiological signals

Physiological signals, also known as vital signs, are measures of various physiological statistics taken from a patient's body in order to assess the most basic body functions. The vital signs describe patient's overall appearance and are essential part of physiological examination, often providing important initial information that influences the direction of evaluation. They are also a highly reliable predictor of life-threatening clinical events.

There are four vital signs, which are standard in most medical settings:

1. Body temperature
2. Heart rate
3. Blood pressure
4. Respiratory rate

Additional vital signs recently introduced into emergency medicine are: a) pulse oximetry (SpO2), which is a non-invasive method to monitor the oxygenation of a patient's haemoglobin; b) capillary refill, which is the rate at which blood refills empty capillaries; and c) the pain scales which measures a patient's pain intensity or other features. In most medical settings so called "fifth vital sign", most often refers to blood oxygen saturation, also known as pulse oximetry. It describes the percentage measure of blood haemoglobin oxygenation.

Four primary vital signs and pulse oximetry are the subjects to measurements in this research project. These measurements provide the most unique and objective assessment of the patient's clinical condition and are a valuable source of different levels of information. They indicate the level of well-being, severity of illness or may dictate the urgency of a required

intervention. Other vital signs were adopted as part of this research due to their irrelevance to the well-being subject or non-quantifiable nature. Capillary refill rate, which in general is considered as part of the assessment of overall perfusion, is most closely linked to circulatory volume and blood pressure in children. Assessment of pain as a vital sign is still gaining acceptance, with number of proposed scales that are based on self-report, observational (behavioural), or physiological data.

Vital signs data recorded over time as well as theirs noted trends, can provide valuable diagnostic tools. For instance, deteriorating vital signs are an important indicator of a deteriorating physiologic condition, whereas improving values provide reassurance that an unstable patient is responding to therapy. Especially when a patient undergoes treatment over an extended period of time, selected vital signs, particularly previously abnormal ones, should be monitored.

Depending on the circumstances, dictated by the patient's clinical state or with any sudden change in the patient's clinical status, vital signs should be measured and recorded at intervals or continuously. An abnormal vital sign can direct the clinician toward a group of diagnoses or a particular organ system for further evaluation. For these reasons, accurate determination, using accurate technique and valid tools as well as further interpretation of vital signs are mandatory. Lack of such examination can lead to delayed diagnosis or misinterpretation of the severity of an illness or injury (Gorgas 2004).

Resting vital signs accepted ranges varies with age group and such must be recognised to enable correct identification of abnormal values and their clinical significance. Normal ranges for vital signs also may be influenced by sex, race, pregnancy, and residence in an industrialised nation. An example of such generally accepted vital signs ranges for adults are, reported in Table 2.1, Modified Early Warning Scores (MEWS) (Subbe, Kruger, Rutherford *et al.* 2001) used in medical admission to hospital. However these ranges have not been validated in patients, who may have had many reasons for vital sign abnormalities, including anxiety; pain; and other forms of distress, in addition to altered physiology from disease states. In these cases individual approach to patient, which will include mental and environmental conditions such as stress or activity is required.

Parameter	Score						
	3	2	1	0	1	2	3
Systolic BP (mmHg)	< 70	71-80	81-100	101-199	-	≥ 200	-
Heart Rate (bpm)	-	< 40	41-50	51-100	101-110	111-129	≥ 130
Respiratory Rate (bpm)	-	< 9	-	9-14	15-20	21-29	≥ 30
Temperature (°C)	-	< 35	-	35-38.4	-	≥ 38.5	-

Table 3.1 Selected vital sign parameters and their significance scores in hospital admission (from Modified Early Warning Scores (Subbe, Kruger, Rutherford *et al.* 2001)).

3.3 Body temperature monitoring

3.3.1 Thermoregulation

Body temperature is a measure of the body's ability to generate and release the heat. Its value greatly depends upon the part of the body at which the measurement is made, the time of a day or level of activity of the person among other variables such as environmental factors. The reference point for body temperature measurement is the core body temperature. It is the operating temperature of an organism, specifically in deep structures of the body such as the liver. Under normal conditions, the core body temperature of healthy, resting adult human remains at 37 ± 0.6°C (98.6 ± 1.08°F). Core body temperature can be maintained within a narrow range, while environmental temperature varies from as much as 13 to 60°C (55 to 140°F) (Cranston, Gerbrandy and Snell 1954). Temperature of peripheral tissues rises and falls with environmental and other influences.

Maintenance of normal constant body temperature requires a balance of heat production and heat loss. Body temperature control is the responsibility of the preoptic area of the hypothalamus, the processing centre in the brain. It responds to changes in the external temperature detected by temperature receptors in the skin, and receptors to detect changes in the internal temperature in spinal cord, abdominal viscera, and central veins. As a response the hypothalamus triggers changes to the effectors such as sweat glands and muscles to ensure our body temperature remains constant (Kasper, Braunwald, Fauci *et al.* 2008).

When tissues are too cold the blood vessels supplying warm blood to the skin become narrow or constrict. This process is called vasoconstriction and results in decrease in the diameter of blood vessels. This reduces the flow of warm blood near the surface of the skin, and reduces heat loss. Figure 3.1a illustrates vasoconstriction.
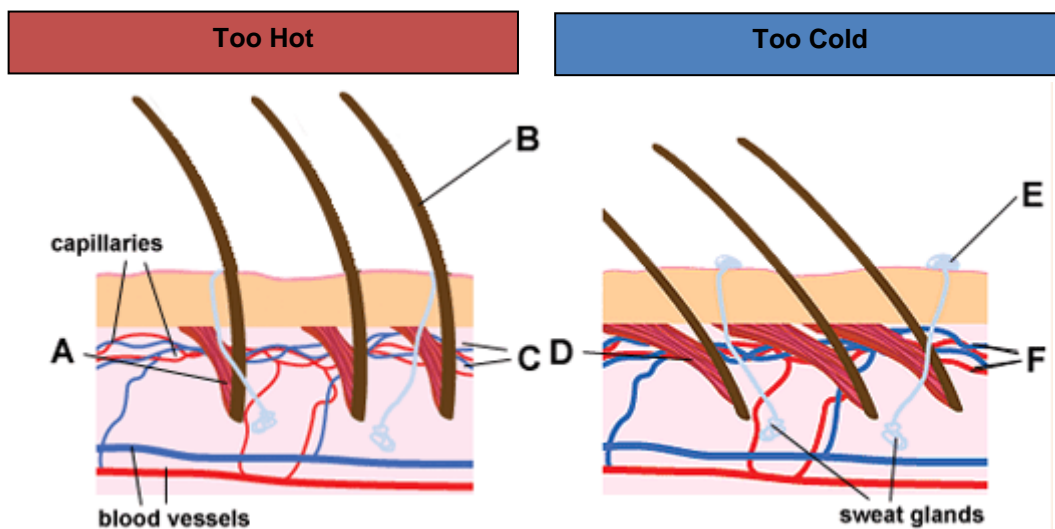
Figure 3.1 a) Vasoconstriction: A - Hair muscles pull hairs on end, B - Erect hairs trap air, C - Blood flow in capillaries decreases; b) Vasodilatation: D - Hair muscles relax. Hairs lie flat so heat can escape, E - Sweat secreted by sweat glands. Cools skin by evaporation, F - Blood flow in capillaries increases (Baker and Driver 2007).

When, in turn, tissues are too hot, blood vessels supplying the blood to the skin begin vasodilatation. Vasodilatation is the process of blood vessels dilatation. This allows more warm blood to flow near the surface of the skin, where the heat can be lost to the air. Heat loss is increased by wind, water, and lack of insulation (e.g., clothing). Figure 3.1b illustrates vasodilatation.

Symptoms as sweating and decreased heat production serve to decrease temperature, whereas shivering, fat catabolism and increased heat production serve to increase body temperature.

3.3.2 Abnormalities

The normal range of human body temperature varies due to an individual's metabolism rate; the higher (faster) it is, the higher the normal body temperature is. Consequently the slower the metabolic rate is, the lower the normal body temperature is. Normal values for body temperature are affected by the following variables: (1) site and methods used for measurement, (2) perfusion, (3) environmental exposure, (4) pregnancy, (5) activity level, and (6) time of day (Gorgas 2004). For instance, the body temperature is lower in the morning, due to the rest the body received and higher at night after a day of muscular activity and after food intake. Normal body temperature may differ as much as 0.8-1.0 °C between individuals or from day to day (Baker and Driver 2007).

3.3.2.1 Hypothermia

Hypothermia occurs when there is a drop in the body's core temperature below 35°C (95°F); this is below the temperature that is required for normal metabolism and body functions. Drop below this temperature normally results in intense shivering, numbness and bluish of the skin, loss of movement of fingers, confusion, sleepiness, depressed reflexes, progressive loss of shivering, slow heart beat and shallow breathing. Drop in core temperature below 32°C (89.6°F) is the medical emergency with hallucinations, delirium, complete confusion, very shallow breathing and slow heart rate which can lead to irregular heartbeat, or respiratory arrest resulting in death (Gorgas 2004, Kasper, Braunwald, Fauci *et al.* 2008).

3.3.2.2 Fever

Fever is a common medical sign characterised by an elevation of temperature above the normal range of 36.5–37.5 °C (98–100 °F) due to an increase in the body temperature regulatory set-point (e.g., from 37°C to 38°C) (Hutchison, Ward, Lacroix *et al.* 2008). For most fevers, body temperature increases by 1°–2°C.

With an increase in person's temperature, there is in general an opposite feeling of cold and shivering. Once the new temperature set-point is reached, there is in general a feeling of warmth. At this point vasoconstriction begins in order to maintain new set-point. A fever is one of the body's immune responses that attempt to neutralize a bacterial or infection. With the

exception of very high temperatures, a controlled fever is normally a good symptom in body infection, which indicates that the immune system is working. Controlled fever is considered as temperature between 37.5–38.3 °C (100–101 °F) (Kasper, Braunwald, Fauci *et al.* 2008)

3.3.2.3 Hyperthermia

Hyperthermia is an uncontrolled fever defined as a temperature greater than 37.5–38.3 °C (100–101°F).   It characterize in an increase in body temperature over the body's thermoregulatory set-point, due to excessive heat production or insufficient thermoregulation. In order to classify the fever as hyperthermia an elevation from the expected temperature must occur, including diurnal variations in temperature which might be as high as 37.7°C (>99.9°F) in late afternoon.

3.3.2.4 Hyperpyrexia

Hyperpyrexia is a fever with an extreme elevation of body temperature greater than or equal to 41.5 °C (106.7 °F) (Kasper, Braunwald, Fauci *et al.* 2008). Such a high temperature is considered a medical emergency and might result in fainting, vomiting, severe headache, dizziness, confusion, hallucinations, delirium, palpitations and breathlessness. Elevation above 43°C is normally a death, but there may be also serious brain damage, continuous convulsions, shock and heat stroke. Cardio-respiratory collapse will likely occur.

	Core Body Temperature
Hypothermia	< 35.0 °C (95.0 °F)
Normal	36.5–37.5 °C (98–100 °F)
Fever	37.5–38.3 °C (100–101 °F)
Hyperthermia	38.4–41.4 °C (101–107 °F)
Hyperpyrexia	> 41.5 °C (107 °F)

Table 3.2 Temperature classification according to (Mackowiak, Wasserman and Levine 1992, Gorgas 2004, Kasper, Braunwald, Fauci *et al.* 2008)

3.3.3   Methods of measurement

The temperature reading greatly depends on the part of the body it is being measured. Measurements are commonly taken in the mouth, the ear, the anus, or the armpit. There are various influencing factors on areas where temperatures are taken.

According to Sund-Levander's et al. (2002) the normal oral temperature, varies between 33.2–38.2 °C. This is due to the fact that oral temperatures are influenced by drinking, eating and breathing. However, this is the accepted standard temperature for the normal core body temperature, however quite not comfortable for long term monitoring.

Rectal temperatures are an internal measurement taken in the rectum, which overall fall into 34.4–37.8 °C. It is the least time consuming and most accurate type of body temperature measurement, being an internal measurement. It is definitely, not the most comfortable method

to measure the body temperature. Moreover, there is lag in rectal temperature measurement behind changes in core body temperature what increase a risk of cross-contamination.

The most non-invasive and most convenient type of temperature measurement is axillary temperature, which is an external measurement taken in the armpit or between two folds of skin on the body. Skin temperatures, measured under the arm or at the forehead, are not always reliable indicators of core body temperature, especially during those critical times when core body temperature is increasing or decreasing. This is because the skin is a tool the body uses to control core body temperature. For example, when fever is increasing people are likely to react by shivering and drawing in heat from the increased core body temperature.

	Men	**Women**	**Overall**
Oral	35.7– 37.7 °C	33.2–38.1 °C	33.2–38.2 °C
Rectal	36.7–37.5 °C	36.8–37.1 °C	34.4–37.8 °C
Tympanic(ear canal)	35.5–37.5 °C	35.7–37.5 °C	35.4–37.8 °C
Axillary (armpit)	-	-	35.5–37.0 °C

Table 3.3 Normal oral, rectal, tympanic and axillary body temperature in adult men and women taken at various body parts (Sund-Levander, Forsberg and Wahren 2002).

Skin temperatures are further influenced by factors such as fever-lowering medication, clothing and external temperature. This is also the longest way of measuring body temperature. Normal temperature for this type of measurement varies between 35.5–37.0 °C.

3.4   Heart monitoring

3.4.1   Basic anatomy of human heart

The heart, in brief, is a pump that receives blood from venous blood vessels at a low pressure, imparts energy to the blood through raising it to a higher pressure by contracting around the blood within the cardiac chambers, and then ejects the blood into the arterial blood vessels. Heart sits in the centre of the cardiovascular system, where its right side pumps deoxygenated blood through pulmonary artery to pulmonary circulation and left side pumps oxygenated blood through aorta to a systemic circulation.

As illustrated in Figure 3.2, the heart consists of four chambers: right atrium (RA), right ventricle (RV), left atrium (LA), and left ventricle (LV). The right atrium receives blood from the superior (SVC) and inferior vena cavae (IVC), which carry blood returning from the systemic circulation. Blood passes from the RA into the right ventricle (RV), which ejects the blood into the pulmonary artery (PA). Blood returns to the heart from the lungs via four pulmonary veins that enter the left atrium and fills the left ventricle (LV). From there it is ejected into the aorta (A) for distribution to the different organs of the body (Klabunde 2005).

The cardiovascular system must be able to adapt to changing conditions and demands of the body. For example, when a person exercises, increased metabolic activity of contracting

skeletal muscle requires large increases in oxygen supply and enhanced removal of carbon dioxide. Another example of adaptation occurs when a person stands up. Gravitational forces cause blood to pool in the legs when a person assumes an upright body posture. In order to prevent this from happening, the control mechanism of heart contraction maintains appropriate heart rate and normal arterial blood pressure. Electrical changes within the myocytes initiate this contraction.

Figure 3.2 Blood flow within the heart (Klabunde 2005)

3.4.2   Electrical activity of the heart

Control of cardiac function is achieved by cells within the sinoatrial (SA) node, located within the posterior wall of the right atrium. SA nodal pacemaker activity constitutes the primary pacemaker site within the heart. SA generates the electrical potential which initiates heart myocytes to contract. As illustrated in Figure 3.3a the action potentials generated by the SA node spread throughout the atria through cell-to-cell conduction. Action potentials in the atrial muscle have a conduction velocity of about 0.5 m/sec.

Non-conducting connective tissue separates the atria from the ventricles. Therefore, as illustrated in Figure 3.3b, action potentials have only one pathway available to enter the ventricles, which is a specialised region called the atrioventrictular (AV) node. The AV node slows down the impulse conduction velocity to about 0.05 m/sec. The delay in conduction between the atria and ventricles at the AV node is important as it allows sufficient time for complete atrial depolarisation, contraction, and emptying of atrial blood into the ventricles prior to ventricular depolarisation and contraction.

Action potentials leaving the AV node enter the base of the ventricle at the bundle of His as presented in Figure 3.3c. These potentials are then propagated on the left and right bundle branches with the high conduction velocity of about 2m/sec. The bundle branches divide into an extensive system of Purkinje fibers that conduct the impulses throughout the ventricles at high velocity of about 4 m/sec. The Purkinje fiber cells connect with ventricular myocytes, which

become the final pathway for cell-to-cell conduction within the heart. After a short time ventricular muscle cells repolarise and the whole cardiac cycle starts again.
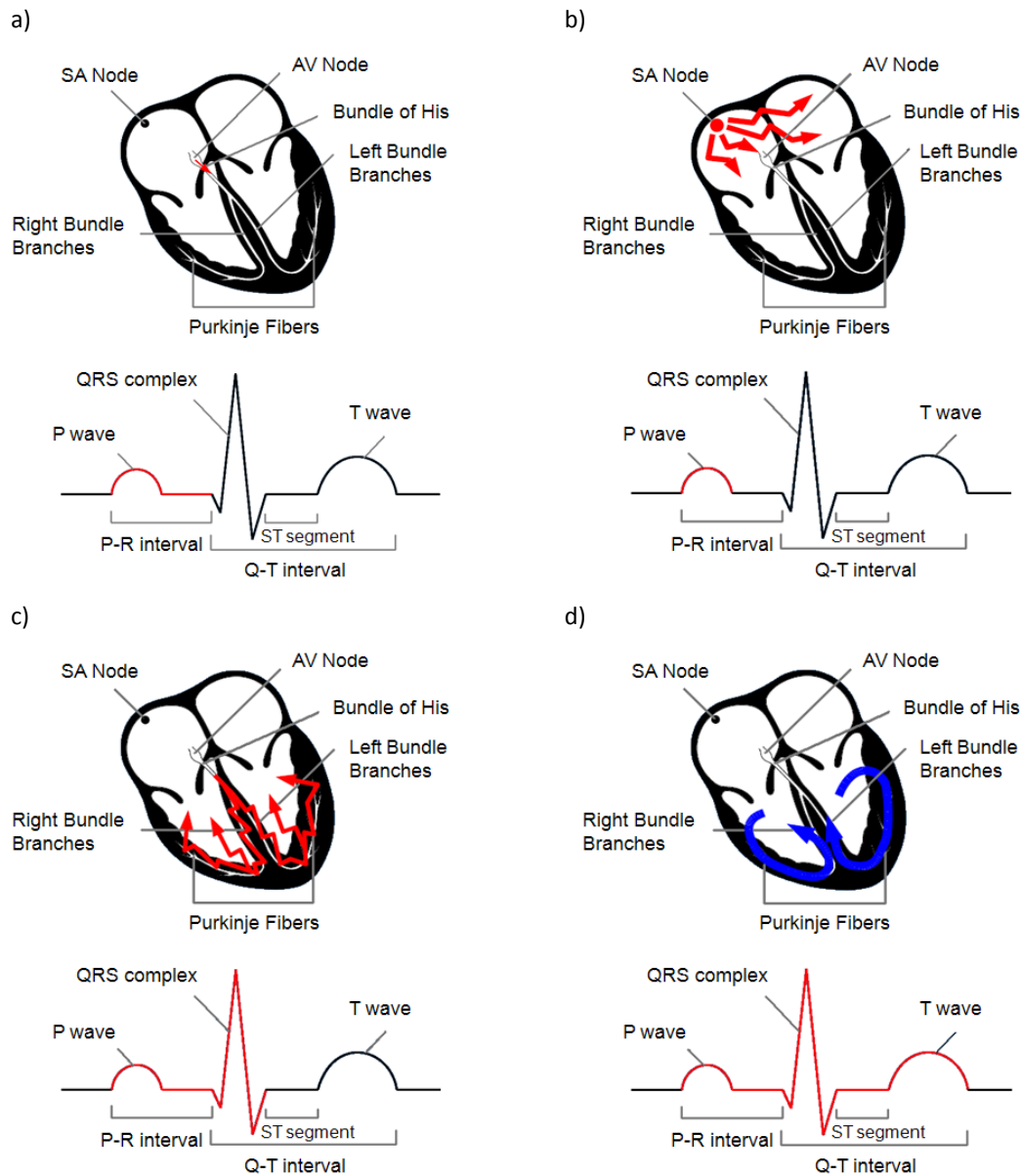
Figure 3.3 Electrical activity of heart.

3.4.3   Method of measurement

The electrocardiogram (ECG) is diagnostic tool that monitors electrical activity of heart measured by an array of electrodes placed at specific locations on the body surface. Electrodes detect the electrical currents that spread throughout the body due cardiac cells depolarisation and repolarisation.

The most basic cardiac monitoring system consists of 3-lead, usually coloured white, black and red, connected to three electrodes adhesed to the patient's chest to form an Einthoven's triangle, as shown in Figure 3.4. There are two colouring schemes: American Heart Association (AHA) standard commonly used in North America and International Electrotechnical Commission (IEC) standard used in Europe (see Table 3.4 below).

3.4.3.1 Placement of ECG Electrodes

W - white lead, always negative polarity

B - black lead, positive for lead I, negative for lead II

R - red lead, always positive polarity
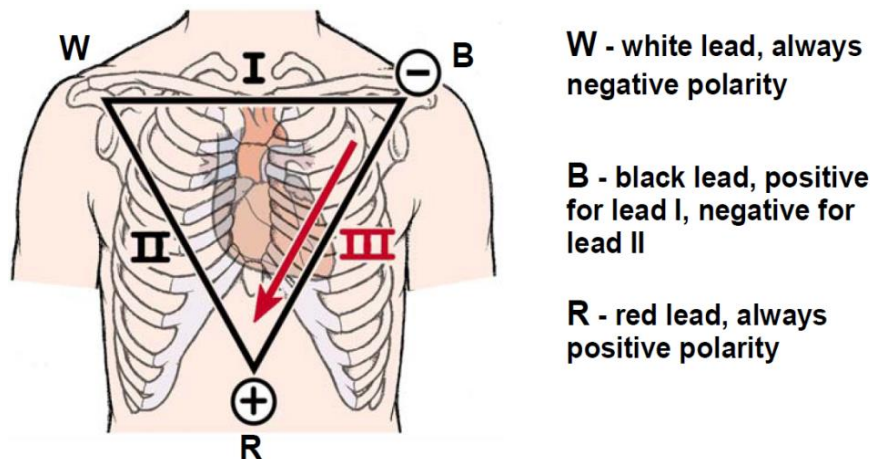
Figure 3.4 The 3-lead cardiac monitoring system (Barill 2005)

AHA (North America)			IEC (Europe)	
Inscription	Colour	Location	Colour	Inscription
RA	White	Right Arm	Red	R
LA	Black	Left Arm	Yellow	L
RL	Green	Right Leg	Black	N
LL	Red	Left Leg	Green	F

Table 3.4 3-lead electrode location standard of the AHA and the IEC

Lead I has the positive electrode on the left arm and the negative electrode on the right arm, therefore measuring the potential difference across the chest between the two arms. In this and the other two limb leads, an electrode on the right leg is a reference electrode for recording purposes. In the lead II configuration, the positive electrode is on the left leg and the negative electrode is on the right arm. Lead III has the positive electrode on the left leg and the negative electrode on the left arm.

Whether the limb leads are attached to the end of the limb (wrists and ankles) or at the origin of the limbs (shoulder and upper thigh) makes virtually no difference in the recording because the limb can be viewed as a wire conductor originating from a point on the trunk of the body. Therefore we place one electrode (RA/White) below the right clavicle, the second (LA/Black) below the left clavicle, and the third (LL/Red) just below the left pectoral muscle. The right leg electrode (RL/Green) located just below the right pectoral muscle is used as a ground (Klabunde 2005).

3.4.3.2　ECG components

An ECG is composed of a time series, special case stochastic processes with a time domain, which forms various waves and lines order into some repeatable pattern. It is displayed on two-dimensional graph where high represents millivolts and width represents time.

By convention, the first wave of the ECG is the P wave. It represents the wave of depolarisation that spreads from the SA node throughout the atria. It is usually 0.08 to 0.1 seconds in duration. Flat region after the P wave represents the time in which the atrial cells are depolarised and the impulse is travelling within the AV node. The period from the beginning of the P wave to the beginning of the QRS complex, is called the P-R interval and normally ranges from 0.12 to 0.20 seconds. This interval represents the time between the onset of atrial depolarisation and the onset of ventricular depolarisation.
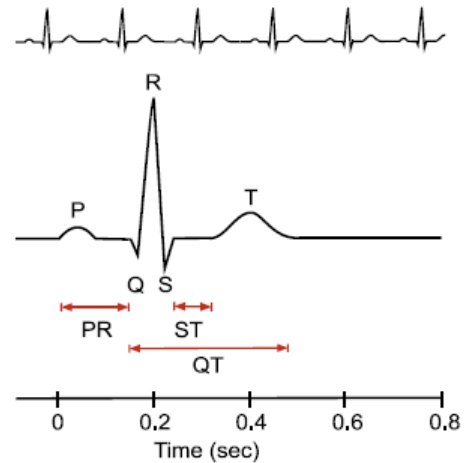
Figure 3.5 Components of the ECG graph.

The QRS complex represents ventricular depolarisation. The duration of the QRS complex is normally 0.06 to 0.1 seconds, indicating that ventricular depolarisation occurs rapidly. Prolonged QRS complex indicate impaired conduction within the ventricles.

The isoelectric period called ST segment, following the QRS complex is the period representing early repolarisation of ventricles. It is a plateau phase, where the cardiac cell membrane potential does not change. Immediately after this the final wave called T wave commence and represents complete ventricular repolarisation.

QRS complex, ST segment and T wave together constitute the Q-T interval, where both ventricular depolarisation and repolarisation occur. This interval roughly estimates the duration of ventricular action potentials. The Q-T interval can range from 0.2 to 0.4 seconds depending on heart rate. For instance, at high heart rates, ventricular action potentials are shorter, decreasing the Q-T interval.

ECG Component	Represents	Normal Duration(sec)
P wave	Atrial depolarisation	0.08 – 0.10
QRS complex	Ventricular depolarisation	0.06 – 0.10
T wave	Ventricular repolarisation	n/m
P-R interval	Atrial depolarisation and AV nodal delay	0.12 – 0.20
ST segment	Isoelectric period of depolarized ventricles	n/m
Q-T interval	Length of depolarisation plus repolarisation (corresponds to action potential duration)	0.20 – 0.40

Table 3.5 Summary of ECG waves, intervals and segments (n/m – not measured) (Klabunde 2005).

3.4.3.3 Heart Rate

Heart rate (HR) is the number of beats the heart makes in a minute and is represented in beat-per-minute (bpm) units. Corresponding to the ECG graph heart rate is the number of QRS complexes, which is the number of ventricular depolarisations in a minute long period. Heart rate is not always the same as pulse. Heart rate is a measurement of electrical activity, while pulse ensures the perfusion of the blood to the target tissues.

Heart rate is the function of time. There are multiple methods to measure the heart rate from an ECG. The simplest, most common and accurate method involves multiplying the number of QRS complexes found over six seconds by a factor of 10 to get a number of QRS complexes in a minute (Barill 2005)

$$HR = N_{QRS/6s} \cdot 10$$

(2)

This method is called the six second count and works well whether the rhythm is regular or irregular as well as for very slow rhythms. Additional increase in accuracy of measurement can be obtained with sliding window.

3.4.4   Normal and abnormal cardiac rhythms

Primary assessment, according to sample algorithm of intervention adapted from Bickley and Szilagyi (Bickley and Szilagyi 2007) shown in Figure 3.6, is to classify the cardiac rhythms as regular or irregular. The point of reference in the assessment of cardiac rhythms, is the resting heart rate monitoring (HR_{rest}). It is a person's heart rate when being at rest. Typical healthy resting heart rate in adults aged 18 and over is 60–100 bpm. Rates below 60 bpm are referred to as bradycardia and rates above 100 bpm are referred to as tachycardia (Klabunde 2005).

	Resting Heart Rate
Bradycardia	< 60 bpm
Normal	60 – 100 bpm
Tachycardia	> 100 bpm

Table 3.6 Resting heart rate classification in adults aged 18 and over

When rhythms are irregular and heart rate is fast or slow, an ECG is required to identify the origin of the arrhythmias, whether it is SN node, AV node, atrium or ventricle, as well as the pattern of conduction. Atrial and ventricular rates and rhythms of depolarisation can be determined from the frequency and amplitude of P waves, QRS complexes and T waves. Table 3.7 outlines the parameters that define the normal and abnormal ECG components. These are accompanied by an incomplete list of some possible cause of abnormalities. In determining abnormalities, in order to trigger warning events, the main focus should be paid to normal parameters and search for any deviations from normality.

Figure 3.6 Algorithm of intervention in cardiac rhythms classification(Bickley and Szilagyi 2007).

Note that amplitudes, presented in the table below, might vary with the actual ECG graph due to the scaling of the signal by an amplification factor. However, the graph calibration ratio between time and voltage lines of 1:2.5 [sec/mV] should be preserved.

ECG Components	Normal Parameters	Abnormal parameters	Some Possible Cause of Abnormal Parameters
P Wave	Upright (positive) Duration: < 0.11 sec Amplitude: < 0.25 mV	Inverted Notched or tall	Junctional rhythm Atrial rhythm, atrial hypertrophy
PR Interval	Duration: 0.12 - 0.20 sec	Duration: Shorter or longer than normal	Junctional rhythm
QRS Complex	Upright(positive), inverted or biphasic waveform Duration: < 0.11 sec Amplitude: > 0.1 mV	Duration: > 0.11 sec	Bundle branch block
QT Interval	Duration: less than ½ the width of R-R interval	Duration: at least ½ of the R-R interval	Long QT syndrome, cardiac drugs, hypothermia Short QT associated with hypercalcemia
ST Segment	In line with baseline (0V) Duration: shortens with increased heart rate	Deviation of 0.05 mV or more from baseline	Cardiac ischemia or infraction, early repolarisation, ventricular hypertrophy
T Wave	Upright (positive), asymmetrical and rounded Duration: 0.10 - 0.25 sec Amplitude: < 0.5 mV	Peaked, inverted, biphasic, notched, flat or wide waveform	Cardiac ischemia or infraction, left bundle branch block, hyperkalemia

Table 3.7 Normal and abnormal parameters of ECG components base on (Barill 2005).

3.5 Blood pressure

3.5.1 Physiology of blood pressure

Blood pressure (BP) is the pressure or force exerted by blood on the walls of blood vessels. It is often referred to as arterial blood pressure (ABP), which indicates blood pressure measured at arterial lines. Each ejection of blood into the aorta by the left heart ventricle results in a characteristic aortic pressure pulse, illustrated in Figure 3.7. Pressures in the cardiovascular system are expressed in millimetres of mercury (mm Hg) above atmospheric pressure.

a)

b)

These images have been removed

The peak pressure of the aortic pulse is called the systolic pressure (SBP), and the lowest pressure in the aorta, which is found just before the ventricle ejects blood into the aorta, is called the diastolic pressure (DBP). The difference between the systolic and diastolic pressures is the aortic pulse pressure. Pulse pressure peak location differs at different parts of the body. As the pressure pulse moves away from the heart, the systolic pressure rises, and the diastolic pressure falls.

Figure 3.7 a) Pressure pulse within the aorta (Klabunde 2005); b) Korotkoff sounds with systolic and diastolic sounds(Bickley and Szilagyi 2007).

As blood is pumped into the resistance network of systemic circulation (ventricles in the body), pressure is generated within the circulatory system. In order to assess systemic vascular function the mean arterial pressure (MAP) is needed which is defined as the average arterial pressure during a single cardiac cycle. At normal resting heart rates, MAP can be estimated from the diastolic (P_{dias}) and systolic (P_{sys}) pressures as:

$$MAP \cong P_{dias} + \tfrac{1}{3}\left(P_{sys} - P_{dias}\right)$$

(3)

It is considered to be the perfusion pressure seen by organs in the body. Mean arterial blood pressure is valuable primary index for continuous blood pressure monitoring, which can trigger the use of the non-invasive cuff measurement, when major BP fluctuations are detected (Fung, Dumont, Ries *et al.* 2004).

3.5.2   Methods of measurement

The blood pressure measurement can be either invasive or non-invasive. Arterial pressures are usually measured non-invasively on skin or rarely invasively, by penetrating the arterial wall to take the pressure measurement.

Traditional non-invasive (indirect) blood pressure measurement consists of deriving the systolic and diastolic blood pressures using an arm cuff and utilizing method of Korotkoff. Other cuff methods make use of pressure measurement in an oscillometry measurement system. The measurement requires the patient to be either in sitting or lying position, as long as the site of measurement is at the level of the right atrium and the arm is supported. The arm cuff is applied on the arm above the elbow, and inflated to about 30 mm Hg above the level at which a palpable pulse disappears. The measurement is taken by deflating the arm cuff at 2 to 3 mmHg per second. The systolic arterial blood pressure is defined as the first appearance of faint, clear, tapping Korotkoff sounds that gradually increase in intensity, whereas the diastolic blood pressure is defined as the point at which sounds disappear (Gorgas 2004). The cuff is not the most suitable method for wearable application, as this implies that complex electronic and mechanical components have to be employed with pressure sensors, which need to detect signals that fall in the range of millivolts. Interesting new cuff less method, suitable for continuous monitoring, includes the pulse transit time (PTT) defined as the time taken for pulsed blood, which is initiated from the heart, to travel to other parts of the human body. Pulse is taken at i.e., finger, ear or toe using pulse oximeter. The PTT is then used to infer the systolic blood pressure, which provides enough information for decision of hypertension and hypotension (Fung, Dumont, Ries *et al.* 2004).

Invasive (direct) measurement involves direct measurement of arterial pressure by placing a cannula needle in an artery line. The cannula needle is then connected through a sterile, fluid-filled system to an electronic pressure transducer that allows displaying a graph of beat-by-beat pressure against time. This method requires very close supervision as might be associated with complications such as infections or bleedings and its use is normally limited to ICU units.

3.5.3   Blood pressure classification

According to the American Heart Association (AHA) (Chobanian, Bakris, Black *et al.* 2003) the normal value for arterial blood pressure is less than 120/80 mmHg, systolic and diastolic respectively. As this is the optimal value it is also a subject to multiple variations, including circadian rhythms, age, postural change, physical activity, food intake etc. (see Figure 3.8).

According to Table 3.8 which provides a classification of blood pressure for adults aged 18 or older, there are two major abnormalities - hypotension, which is an abnormally low BP,

and hypertension, which is an abnormally high BP. Prehypertension, included in new American Heart Association BP classification, introduced with the seventh report of the Joint National Committee on prevention, detection, evaluation, and treatment of high blood pressure, is not a disease category. Rather it is a designation chosen to identify individuals at high risk of developing hypertension, so that both patients and clinicians are alerted to this risk and encouraged to intervene and prevent or delay the disease from developing. Individuals who are prehypertensive are not candidates for drug therapy on the basis of their level of BP and should be firmly and unambiguously advised to practice lifestyle modification in order to reduce their risk of developing hypertension in the future.

This table has been removed due to third party copyright.

Table 3.8 Blood pressure classification according to AHA (Chobanian, Bakris, Black *et al.* 2003)

Hypotension is generally considered as systolic blood pressure less than 90 mmHg and diastolic less than 60 mmHg. In spite of not being treated as disease but rather as physiological state, hypotension can be linked with various disorders such as reduced cardiac output, widening of blood vessels, medicine side effects, anemia, heart & endocrine problems.

Hypertension is classified as a disease with systolic pressure above 140 and diastolic pressure above 90 mmHg. According to the revised AHA hypertension categories it can be further classified as stage 1 or stage 2 hypertension. This classification suggests that individuals with prolonged stage 1 (140 – 159 / 90 – 99 mmHg) and 2 (≥ 160 / 100 mmHg) hypertensions are candidates for drug therapy. The higher the BP, the greater is the risk of strokes, heart attacks, heart failure and kidney or arterial diseases.

3.5.4   Blood pressure variation

Systolic and diastolic arterial blood pressures are not static but undergo natural variations from one heartbeat to another and throughout the day in a circadian rhythm. They also change in response to stress, nutritional factors, drugs, disease, exercise, and momentarily from standing up. Sometimes these variations are large therefore continuous BP monitoring has to take this variability into account, when attempting its assessment. Figure 3.8 presents the overview of the blood pressure variations and some possible causes of these variations.
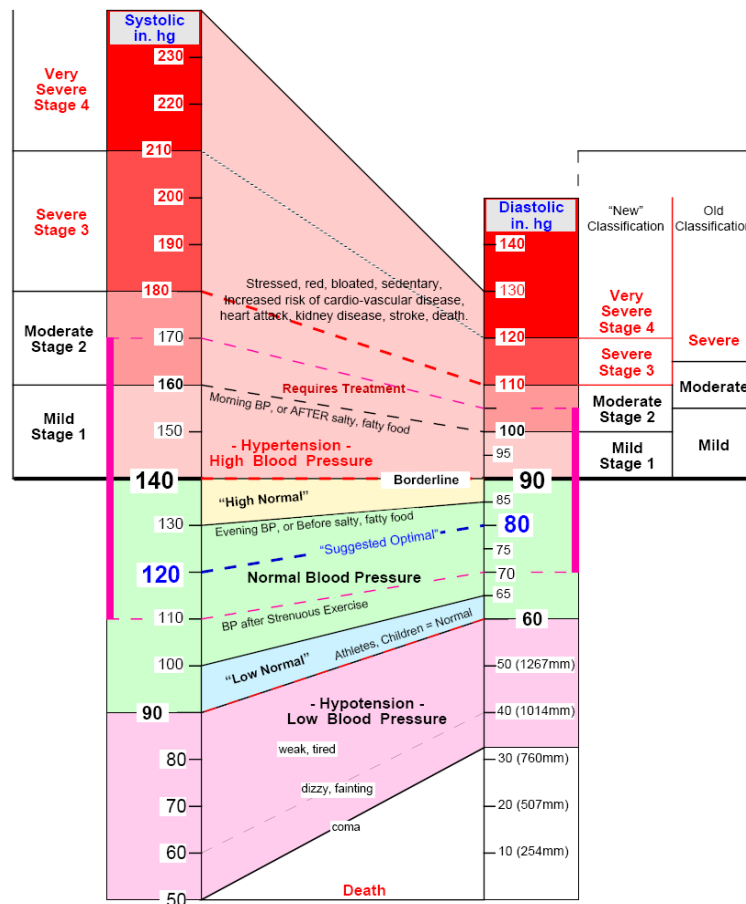
Figure 3.8 Blood pressure variation chart

3.6   Blood oxygen saturation

3.6.1   Physiology of blood oxygen saturation

Blood oxygen saturation, also called pulse oximetry or simply oximetry (SpO_2), is the measurement of the percentage saturation of oxygen in haemoglobin. It is a determinant of how well oxygen is supplied to the body tissues. If the oxygen is not delivered properly, cell tissues will be damaged. It is important to note that pulse oximetry is used to monitor oxygenation, but cannot determine the metabolism of oxygen, or the amount of oxygen being used by a patient.

The basic concept in oximetry is to transmit light through a blood sample. For this purpose a digital photoplethysmograph (DPP) is used. It is an optically obtained volumetric measurement of an organ (Shelley 2007). A DPP is obtained by illuminating the skin and measuring changes in light absorption. The blood will absorb some amount of light emitted, depending on the concentration of oxygenated and deoxygenated haemoglobin. The intensity of light transmitted across the cell tissue varies as shown in Figure 3.9b. Such pulsate variation of the signal in time is due to the heart beat, which generates the systolic peak in the vertices. The amplitude of this cardiac−synchronous pulsate signal is approximately 1% of the DC level on

which the signal is superimposed. Only that part of the signal, which is directly related to the inflow of arterial blood into the body segment, is used for the calculation of oxygen saturation.

3.6.2 Method of measurement

Pulse oximetry uses two light diodes (LEDs) emitting red and infrared wavelengths. These wavelengths yield the best results having the greatest separation between the haemoglobin and oxyhaemoglobin absorption spectra. The measurement assumes that blood is composed of only oxygenated and deoxygenated haemoglobin.

a)

These images have been removed

The attenuation of light by the body segment, normally finger, toe or ear-lobe, can be split into the three independent components, shown in Figure 3.9a. The light illuminates both arterial (A) and venous (V) blood and must traverse all tissues (T) between light source and receiver. Arterial blood absorption divides further on pulsating, which is the variable AC path length, and non-pulsating. Tissues absorption, venous and capillary blood absorption as well as non-pulsating arterial blood absorption forms the DC path length.

b)

Figure 3.9 a) Light path length in time, b) Typical pulsatile signal (Townsend 2001).

To calculate the pulse oximetry (SpO_2), from photoplethysmograph presented in Figure 3.10, use two equations which utilize the AC and DC components. The first step is to use the red and infrared signal to calculate an R value, which is the normalised ration of red to infrared transmitted light intensity, given in equation (4).

$$R = \frac{AC_{red} / DC_{red}}{AC_{inf} / DC_{inf}}$$

(4)

The second step is to use a linear approximation to calculate the final value of SpO_2, as given in equation (5).

$$SpO_2 = 110 - 25R$$

(5)

This empirical approximation corrects an error, which is a result of assumption the presence of only two substances in the light path. This assumption in fact is wrong, as several other substances exists, however it is satisfactory for the purpose of pulse oximetry (Rusch, Sankar and Scharf 1996).
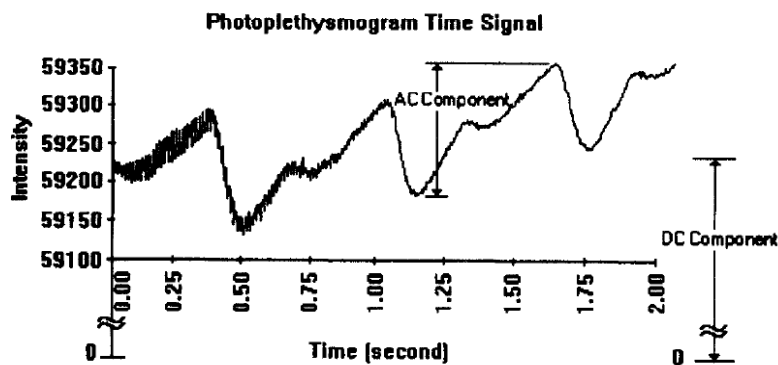
Figure 3.10 Pulse oximetry signal with pulsating component.

3.6.3   Normal SpO$_2$ values

Under normal physiological conditions when a patient is breathing room air, SpO$_2$ of more than 95% is regarded as normal. A value of 100% is not normally observed when breathing air unless the subject is hyperventilating. It might be observed also in a patient on a supplementary O$_2$, what normally indicate that the patient is receiving too much O$_2$. Moreover, saturation of 100% may compensate for problems of O$_2$ carriage, such as anaemia.

When the SpO$_2$ falls to 90% we can say about the respiratory failure. However, this is not a stiff rule as some patients with chronic respiratory disease may tolerate lower saturations. Therefore, when interpreting the pulse oximetry readings it is important to take other vital signs into account as well as review the history of the patient case (Kendrick 2008).

3.7   Respiratory rate

3.7.1   Physiology of respiratory system

The respiratory system's major functions are: a) to provide an adequate oxygen (O$_2$) supply to meet the energy production requirements of the body, and b) maintain a suitable acid-base status by removing carbon dioxide (CO$_2$) from the body. This is accomplished by moving volumes of air into and out of the lungs and is accompanied with expansions and contraction of the chest cage. This act of normal breathing has a relatively constant rate and inspiratory volume that together constitute normal respiratory rhythm (Braun 1990).

During normal respiration, the lungs exchange about 500 ml of air 12 times a minute. This is the tidal volume of air inspired or expired. Tidal volume exchange is the main determinant of respiratory rate. In young adult males, there is an inspiratory reserve volume of about 3000 ml that can be inspired above the tidal volume, while the expiratory reserve volume is about 1100 ml, which can be forcefully expired. The vital capacity, volume which can be exhaled after maximal inspiration, includes all of the above yielding about 4600 ml. Well-trained athletes may have values 30-40% higher, while females generally have 20-25% less for the quantities listed

above. The residual volume represents the amount of air that cannot be expelled even by forceful expiration and averages about 1200 ml. It prevents lungs from totally collapsing and expelling all air, as it requires too much energy to reinflate them from the collapsed state. The functional residual capacity is the amount of air that remains behind during normal breathing, which amounts to 2300 ml. Figure 3.11 shows respiratory excursion during normal breathing, where lung capacities and volumes were marked (Guyton and Hall 2006).

Figure 3.11 Respiratory excursions during normal breathing and during maximal inspiration and maximal expiration adapted from (Guyton and Hall 2006).

3.7.2   Methods of measurement

Simple respiratory rate measurement method base on chest and abdomen observation, which moves with each inspiration and expiration as presented in Figure 3.12. Physiological examination settings require examiner to count the number of times the abdomen or chest wall

Figure 3.12 Chest wall and abdominal coordination during tidal breathing (Braun 1990).

rises for 15 seconds and multiply by 4, or for 30 seconds and multiply by 2. For greater accuracy when the rate is very low, count for full minute will give best results. Measurement should be performed without patient being aware of measurement as it might significantly affect the result.

Continues respiratory rate measurement systems most commonly relay on piezoelectric , which measure chest and abdominal expansion, intranasal pressure transducers or nasal

thermocouple, which monitor airflow by detecting pressure or temperature changes respectively (Marks, South and Carter 1995). Such systems produce a small voltage when stress on the piezoelectric is applied or change its resistance when airflow change in temperature or pressure is detected. When using piezoelectric sensor there is an advantage that no interface or external power source is necessary since it produce its own signal. On each breath the sensor's voltage is amplified and filtered to produce a clean, reliable respiratory signal.

3.7.3   Abnormalities in rate and rhythm of breathing

Under normal physiological conditions when a patient is in rest and is breathing room air, his/her optimal respiratory rate should remain at constant average level of 12 breaths per minute. In most individuals, the rate and pattern are surprisingly constant, only interrupted every several minutes by a larger inspiratory effort or sigh. As literature shows, estimates of respiratory rate do vary between sources from 12 up to 20 breaths per minute and as such are treated as normal (accepted) values (Subbe, Davies, Williams *et al.*).

The definition of an abnormal respiratory rate for adults varies from over 14 to over 36 breaths/minute. Recent evidence suggest that an adult with respiratory rate of over 20 breaths/minute is probably unwell and an adult with respiratory rate of over 24 breaths/minute is likely to be critically ill (Cretikos, Bellomo, Hillman *et al.*). Abnormally fast breathing is called tahypnea and is defined as reparatory rate over 20 breaths/minute. The opposite of tahypnea is called bradypnea and is defined as respiratory rate below 12 breaths/minute.

Recognising alterations in respiratory rate is an important early blue of disease recognition. While frequently it is nonspecific, in many cases it can lead directly to a diagnosis.

Figure 3.13 Abnormalities in rate and rhythm of breathing adopted from (Bickley and Szilagyi 2007).

3.8   Conclusions

Vital sign recording, which are the primary reference point for the most basic body functions, are still rarely stored anywhere other than on a paper chart or Intensive Care Unit (ICU) monitors in spite of being a valuable source of different levels of information. This data recorded over time as well as theirs noted trends, can provide valuable diagnostic tools. First, the records can be used to verify whether patient is alive and his life is not threatened.

Moreover, continuous monitoring, for instance prior to hospital admission, can supply valuable information on current patient's health conditions as well as provides an insight on the patient's history in order to help to identify any abnormalities. The other possible medical applications are: diagnostic procedures, emergency response at the disaster scene and ambulatory settings, optimal maintenance of chronic conditions, monitoring the adherence to treatment guidelines or the effects of drug therapy. Apart of this, continuous vital signs monitoring is a valuable method in sport - to assess athletes performance, in military - to monitor soldiers in action, or at the disaster scene - to monitor selected parameters during work in fire-fighter's' clothing amongst others.

Further research presented in this work focus on ubiquities vital signs monitoring and processing. This data analysis sets out the reference point for intelligent and portable vital signs processing algorithms that could closely monitor changes in user's vital signs and maintain an optimal health status independently from the context of use. Acquisition of many different parameters, their integrated processing and continuous adaptation of the interpretation depth allows us to follow the patient's state and diagnosis goals in various situations. Providing this highly adaptable system, by designing intelligent algorithms, capable of self-learning from a set of training data, such as historical data describing a particular relationship or expert opinions, would help in keeping patients physically active, providing early notification and detection of potential illness risks and finally more accurate treatment (Hass and Burnham 2008). Moreover, by integrating this system into wider telemedical framework, it can alert to the medical personnel or emergency services when life-threatening changes occur.

Chapter

4

Portable vital signs monitoring framework: System design

4.1 Introduction

First step towards a fully operational, intelligent well-being monitoring system is a development of hardware and software platform, where the system will reside. Design of such framework must foremost answer user's requirements in terms of usability, reliability, comfort of use and security. The system must fulfil some requirements in order to provide external services for the user or third party users such as medical professionals or emergency caregivers. The hardware should be as little intrusive as possible, small and safe in order to be worn with ease. The software design must be flexible enough to allow for rapid future system extension, customisation and adaptation to continuously changing conditions. These and other requirements are crucial for good system design. The following chapter looks in details on system design, guided by requirements analyses. It focuses on high-level system architecture as well as detailed software design.

4.2 Overall architectural requirements analysis

4.2.1 Introduction

This section outlines services that a system must provide and constraints under which it must operate. It focuses on the health and well-being monitoring concept from four main perspectives: user, system, hardware and software as well as operational domain. Each derives functional and non-functional requirements that the system should answers. In order to avoid ambiguity, lack of clarity and repetition, we will use the following language convention: "must" for mandatory requirements, "should" for desirable requirements and "may" for optional requirements as introduced in (Bradner 1997) and used by (Trossen and Pavel 2006).

4.2.2 User requirements definition

User requirements describe both functional and non-functional requirements in a way that they are understandable by the system user, who doesn't need to have a detailed technical knowledge. Functional user requirements are high-level statements of what the system should do, while non-functional user requirements describe constraints on these services such as reliability measures, portability requirements (e.g. hardware size or weight), usability requirements (e.g. methods of user-system communication – touch, voice, etc.) or safety requirements.

4.2.2.1 Functional user requirements

The following section provides a list of high-level statements of what the system should do, annotated where necessary.

F. Req. 1 The system MUST monitor user's physiological signals.

F. Req. 2 The system MUST provide user's vital signs acquisition from the collection of different sensors.

F. Req. 3 The user MUST be able to discover and register sensor devices.

F. Req. 4 The user SHOULD be able to select which vital signs are monitored and whether the monitoring is continuous, periodic or just a single measurement.

 F. Req. 4.1 The user SHOULD be able to view monitored data in real-time.

 F. Req. 4.2 The user MUST be able to retrieve this data in the future.

F. Req. 5 The user MUST have a means of system management through user-friendly personal device.

F. Req. 6 The user MUST be able to initiate and terminate the monitoring.

F. Req. 7 The user MAY be able to customise monitoring goals and plans.

F. Req. 8 The user

 F. Req. 8.1 MAY get a feedback on his/her health or well-being status.

 F. Req. 8.2 *MUST* be informed when any anomalies are detected.

 F. Reg. 8.3 *MAY be* advised in safe and reliable way on any steps he/she should take regarding life style, diet, stress management, physical activity or any medical consultation required.

 F. Req. 8.4 *MUST be* alarmed when any life threatening event occurs.

F. Req. 9 The system SHOULD provide data/information presentation to the user

F. Req. 9.1   The system SHOULD provide simple and intuitive GUI interface.

F. Req. 9.2   The system MAY provide data visualisation mechanisms.

F. Req. 10   The system MAY provide a user with an easy form of voice/text communication with medical experts*.*

The following general use case diagram illustrates main systems functions and their relationships, which shall be available to the user and medical expert.
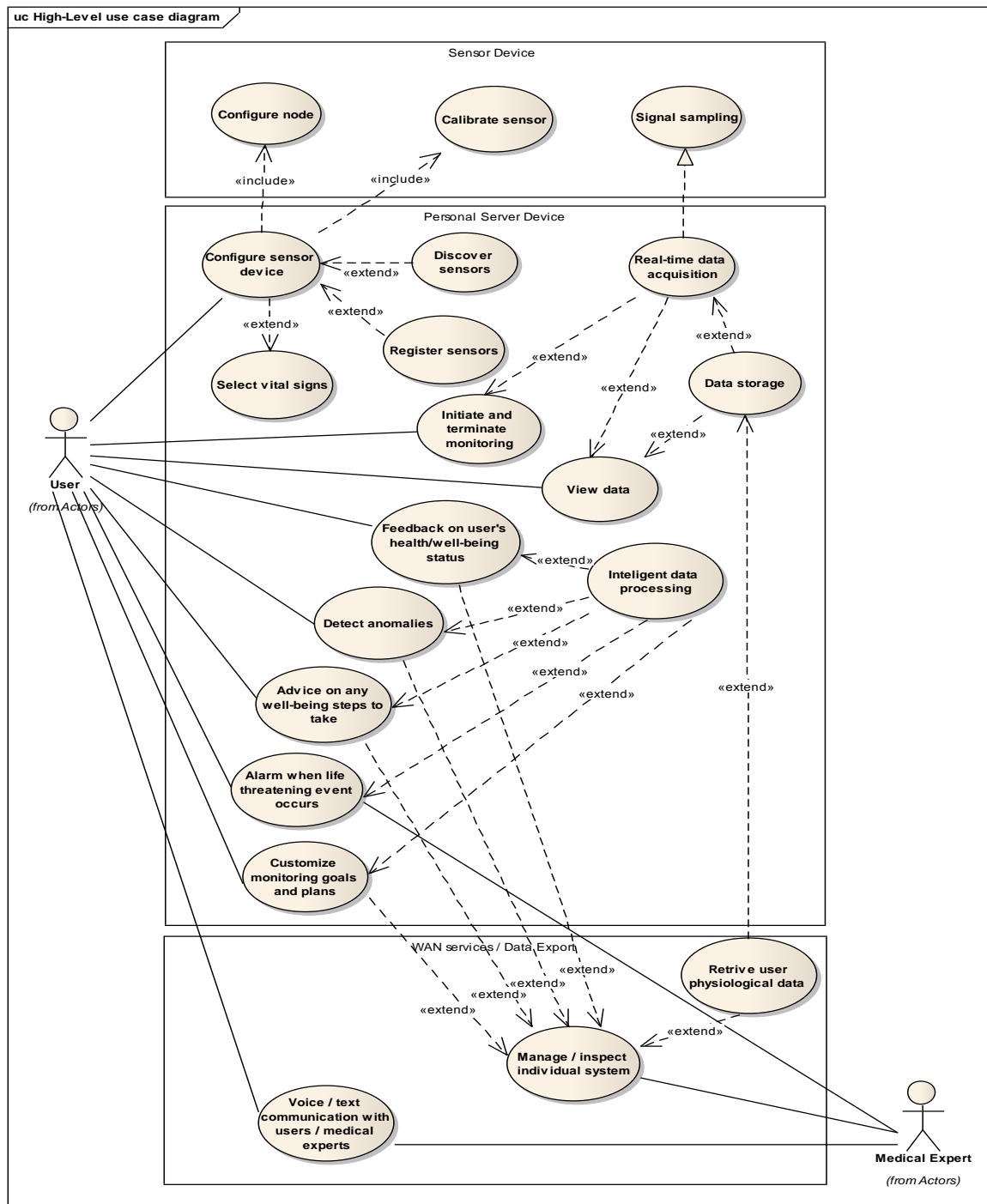
Figure 4.1 High-level perspective system use case diagram

4.2.2.2 Non-functional user requirements

The following section provides a list of user perspective constraints and properties that the system design should meet.

NF. Req. 1 Physiological signals monitoring MUST perform in non-invasive manner.

NF. Req. 2 Monitoring MUST be able to perform continuously (except any assumed situations when it shouldn't).

NF. Req. 3 Sensor devices SHOULD be small, lightweight and wearable.

NF. Req. 4 Sensor devices MUST be wireless and do not limit user movements.

NF. Req. 5 Sensor devices SHOULD be as little intrusive as possible allowing the user to perform normal day-to-day activities.

NF. Req. 6 Sensor devices MAY be resistant to mechanical damages and waterproof.

NF. Req. 7 The system interface SHOULD be clear and intuitive.

NF. Req. 8 The system interface MAY be touch controlled.

NF. Req. 9 The system MAY provide other forms of user communication such as sound or voice notifications.

NF. Req. 10 The system configuration SHOULD be simple and straightforward.

NF. Req. 11 User-system interaction MUST be as automated as possible needing user interaction in critical events.

NF. Req. 12 The user MUST be able to use other personal device's functions such as phone calls, while performing monitoring.

NF. Req. 13 The user SHOULD be able to use any mobile device as personal assisting device as soon as it fulfils application requirements in terms of processor speed, available memory, and communication protocol and storage capacity.

NF. Req. 14 The user's data MUST be kept encrypted and anonymised.

    NF. Req. 14.1 The user's identifiable data MUST not to be kept either on personal server or sensor device.

    NF. Req. 14.2 The user's monitoring records MUST only be de-anonymised by the upload to the medical expert for manual analyses.

    NF. Req. 14.3 The access to the user data and personal server for third parties MUST be password protected through secured connection.

NF. Reg. 15 The user SHOULD be able to choose whether the system should work as a standalone system or as part of a broader telemedical framework.

4.2.3   System requirements specification

System requirements define a detailed description of system services seen from the perspective of the system architect.   It focuses on system components, data model, communication protocols and interfaces to other systems. In the following sections we look at each component's function, how it should interface with other components and what dependences it should implicate. In data model specification we look at what are the data types to store, where and in what form. In communication requirements specification we describe how one component can talk with the others, the protocol it should utilize and levels of communication. Finally, as most systems also this should link with other systems and the operating interfaces must be specified as part of the requirements.

Not all elements described in this section are directly exposed to the user. System requirements specification along with the user requirements definition will be used to draw the concept of the system architecture.

4.2.3.1   General system requirements

F. Req. 10   The system MUST provide personal assisting application deployed for a PDA or smartphone.

  F. Req. 10.1   It MUST support management of system resources.

  F. Req. 10.2   It MUST provide real-time intelligent data processing.

F. Req. 11   The system MUST provide data acquisition by means of distributed sensor nodes with single or multiple sensors.

  F. Req. 11.1   Each sensor node MUST support aggregation of at least five different sensor channels.

  F. Req. 11.2   The system MUST support integration of new sensor nodes from different vendors with different protocols.

F. Req. 12   The system MUST support different communication technologies for personal server within the wireless body area network.

  F. Req. 12.1   It MUST support at least Bluetooth wireless communication protocol.

  F. Req. 12.1   It SHOULD provide a way to integrate ZigBee and proprietary wireless communication protocols.

F. Req. 13   The system SHOULD provide data/information distribution to third parties through Wide Area Network (WAN) services

4.2.3.2 Sensor node requirements

F. Req. 14 The sensor node MUST support data acquisition from on-board sensors.

F. Req. 14.1 It MUST support continues data acquisition.

F. Req. 14.2 It MUST support on demand data acquisition.

F. Req. 15 The sensor node MUST provide

F. Req. 15.1 sensors discovery

F. Req. 15.2 sensors registration with personal device

F. Req. 15.3 sensors calibration
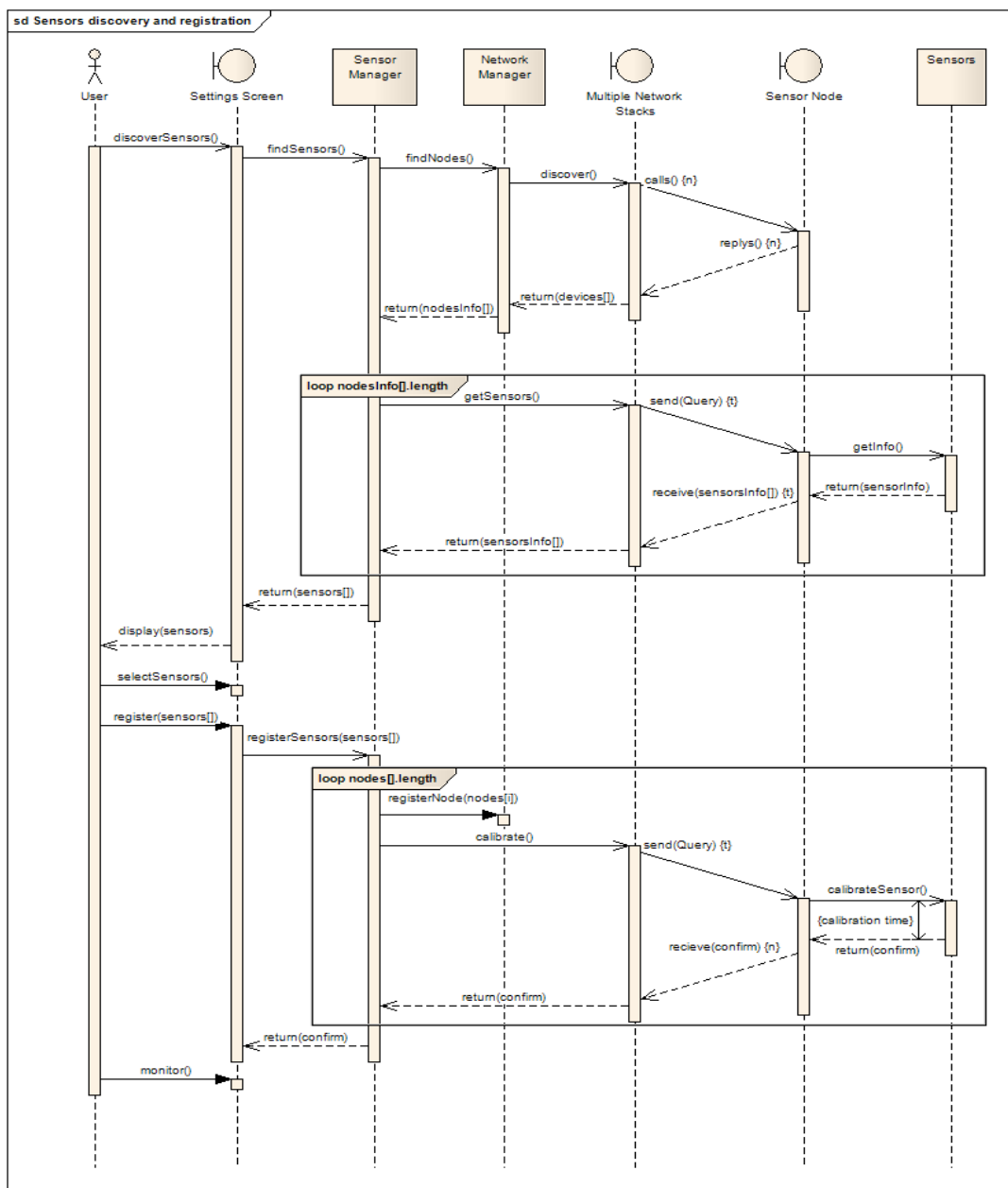
Figure 4.2 Sensor discovery and registration sequence diagram

F. Req. 16   The sensor node MUST support data filtering and pre-processing.

F. Req. 17   The sensor node MUST support data storage.

F. Req. 18   The sensor node MUST support at least Bluetooth wireless communication

> *F. Req. 18.1*   It MUST support synchronous transmission of data packages to personal server.

> *F. Req. 18.2*   It MUST support asynchronous transmission of query/configuration packages from personal sever.

F. Req. 19   The sensor node MUST provide the communication protocol that implements commands and sensor specific queries that the node will reply to.

4.2.3.3   Personal assisting device requirements

F. Req. 20   The server MUST provide remote sensor devices management.

> F. Req. 20.1   It MUST allow discovery of data sources within the network.

> F. Req. 20.2   It MUST allow registration of data sources with personal server.

F. Req. 21   The server MUST support exchange of data/configuration packages with sensor nodes over the network.

> F. Req. 21.1   It MUST accept physiological data transmission from sensor nodes.

> F. Req. 21.2   It MUST support remote sensor nodes' configuration.

> F. Req. 21.3   It MUST support data packages synchronization within WBAN.

F. Req. 22   The server MUST provide a collection of measured physiological data.

> *F. Req. 22.1*   It MUST be possible to obtain a plain physiological data.

> *F. Req. 22.2*   It SHOULD be possible to obtain higher level semantic data.

F. Req. 23   The server MUST support data storage.

> *F. Req. 23.1*   It MUST support text/binary file format.

> *F. Req. 23.2*   It SHOULD support structured XML mark-up language file format.

F. Req. 24   The server MUST support data processing

> *F. Req. 24.1*   It MUST support data aggregation from different sources.

> *F. Req. 24.2*   It MUST provide real-time intelligent data processing runtime.

> *F. Req. 24.3*   It MUST provide event trigger and intelligent decision algorithms in case of emergency situation or abnormalities detection.

> *F. Req. 24.4*   It SHOULD be context-aware.

F. Req. 25   The server SHOULD control efficient data sources utilization

> *F. Req. 25.1*   The data sensing SHOULD be context-aware and use right data sources right when it is needed for processing algorithms.

> *F. Req. 25.2*   Intelligent data processing algorithms SHOULD be able to control data sources usage and reduce WBAN traffic.

F. Req. 26   The server MUST provide data presentation layer

> *F. Req. 26.1*   It SHOULD provide real-time data presentation

> *F. Req. 26.2*   It SHOULD provide data visualisation through

> > *F. Req. 26.2a*   different charts plotting mechanism

> > *F. Req. 26.2b*   simulation mechanisms.

> *F. Req.26.3*   It MUST provide appropriate user friendly GUI controls.

F. Req. 27   The server SHOULD provide data distribution over WAN network.

F. Req. 28   The server SHOULD provide and manage WAN bearers of information.

> *F. Req. 28.1*   It MAY support SMS

> *F. Req. 28.2*   It MAY support voice communication

> *F. Req. 28.3*   It SHOULD support TCP/IP protocol.

> > *F. Req. 28.3a*   It SHOULD support HTTP/S protocol.

> > *F. Req. 28.3b*   It SHOULD support SSH protocol in order to log into a remote machine and execute commands

F. Req. 29   The system SHOULD be fully accessible and configurable remotely by third parties such as medical experts.

F. Req. 30   The server MAY support device embedded services

> *F. Req. 30.1*   It MAY support USB data export.

> *F. Req. 30.2*   It MAY support GPS service for user location.

4.2.4   Domain specific requirements

This section describes system characteristics and features that reflect the medical domain of the application. It defines additional functional requirements, constraints on existing ones and explicit computations that this specific application of the system must satisfy.

F. Req. 31 The system MUST allow for direct sampling of four physiological signals:

- Body temperature (TEMP)
- Electrocardiogram (ECG)
- Respiratory rate (RESP)
- Pulse oximetry (SpO2)

F. Req. 32 The system MUST derive indirect vital signs indicators

F. Req. 32.1 The system MUST perform continuous and non-invasive blood pressure measurement, proposed by Fung et al. (Fung, Dumont, Ries *et al.* 2004) that infer blood pressure from Pulse Transit Time (PTT) using the following equation:

$$BP = \frac{1}{0.7}\left(\frac{1}{2}\rho\frac{d^2}{PTT^2} + \rho g h \right) \tag{6}$$

where ρ = 1035 kg/m3, is the density of blood, d the distance from heart to the other part of body, PTT the pulse transit time, g the gravitational pull and h the height difference between two sites. Figure 4.3 illustrates above algorithm.

Figure 4.3 Continuous blood pressure (BP) measurement by Pulse Transit Time (PTT) (Tay, Guo, Xu *et al.* 2009).

F. Req. 32.2 The system MUST calculate heart rate as a number of beats-per-minute (bpm) from ECG signal. It can be calculated as 60 (number of seconds in a minute) divided by the average R-R interval (in seconds).

$$HR = 60/RR_{interval} \tag{7}$$

Figure 4.4 Calculation of Heart rate (HR) from Electrocardiogram signal (ECG).

F. Req. 33 The obtained data MUST be validated against vital signs medically accepted ranges such as MEWS, the context such as activity performed and other external parameters.

4.3 System architecture

4.3.1 Introduction

A number of patient health monitoring systems that perform monitoring of patient's physiological signals, were proposed and developed in research and commercial projects in recent years. These in general were simple and deterministic monitors that informed medical personnel of patient health conditions. Most of them focus on physiological data acquisition by means of wireless sensors that according to their placement on human body can be classified into two groups a) intra-body sensors (Jones 2006, Valdastri, Rossi, Menciassi *et al.* 2008, Rodrigues, Caldeira and Vaidya 2009) and b) sensors that are placed outside the body, where most of them operates in direct contact with skin (Anliker, Ward, Lukowicz *et al.* 2004, Wu, Bui, Batalin *et al.* 2008, Tay, Guo, Xu *et al.* 2009). Due to non-invasive nature of this project we will focus on the second group of sensors. These include: accelerometers, body temperature, electrocardiogram (ECG), heart rate, blood pressure, blood oxygen saturation (SpO2), respiration, among others. The first part of this section presents some of the previous research efforts in designing wearable physiological parameters monitoring systems and how their influenced development of this proposal. In the second part a description of the proposed system design is provided.

4.3.2 Related works

First research efforts in designing wearable health monitoring systems focused on individual systems capable to sense single or multiple vital signs as well as store and transmit obtained data locally over wireless connection. A few notable very first systems include: UbiMon system (Laerhoven, Lo, Ng *et al.* 2004) capable to monitor patient's ECG and transmit the signal, MobiHealth (Konstantas, Jones, Bults *et al.* 2002) system, that aimed at introducing new mobile services based on 2.5 (GPRS) and 3G (UMTS) technologies, MIThril (Devaul, Sung, Gips *et al.* 2003) a wearable computing platform compatible with both custom and off-the-shelf sensors able to monitor ECG, skin temperature, and galvanic skin response (GSR), AMON (Anliker, Ward, Lukowicz *et al.* 2004), a wrist-worn integrated system applying aggressive low-power design techniques, or CodeBlue (Shnayder, Chen, Lornicz *et al.* 2005), a Harvard University research project, focused on wireless pulse oximetry, ECG and motion, which demonstrated the formation of ad-hoc networks. The architecture of these systems could be classified as two tier that is made up of medical sensors network and PDA/PC that collects data. They primarily focused on physiological data acquisition and do not provide methods that assist medical professionals and the patients with data interpretation or diagnosis in real time.

Noteworthy example of a fully integrated telemedical system architecture that took all relevant parties and services on-board, were outlined by Otto et al.(2006). Proposed system

spanned a three tier network comprised of a) tier 1 - wireless body area sensors network (WBASN), b) tier 2 - individual health monitoring systems and c) tier 3 - Wide Area Network (WAN) connection to a medical server that resides at the top of this hierarchy. Despite of a great improvement of this solution to a data flow between patients and medical personnel this approach simply equipped traditional personal monitoring systems used to collect data with wireless WAN data transmission mechanism. One of the biggest disadvantages of this solution was data processing that performed on the centralised medical server and most often with the help of medical professionals. The medical servers were optimised to service hundreds or thousands of individual users. Its responsibility was to authenticate users, accept health monitoring sessions upload, format and insert the sessions data into corresponding medical records, analyse the data patterns, recognise serious health anomalies in order to contact emergency caregivers or forward new instructions to the user such as prescribed exercises. The multiplicity of parameters being collected by such systems made them impractical for continuous monitoring due to transmission overload, high rate of false alarms, lack of on device adaptation mechanism to changing conditions such as activity performed, static sensing model, inability of early detection of medical disorders due to offline data processing or high cost of service maintenance due to high demand for processing power (Wu, Bui, Batalin *et al.* 2008).

Recent works of Wu et al. (2008), answered some of the drawbacks pointed out in Otto's architecture while designing MEDIC, a medical embedded device for individualised care. The system, which also follows three tier architecture, offered and integrated approach based on standard hardware and software, with emphasis on local signal processing, wireless sensing, high-bandwidth data streaming from multiple wireless sensing devices as well as interaction and control mechanisms for medical personnel and patients. It brought to attention an aspect of sensors adaptation mechanism to optimise the accuracy of diagnostic inference regarding patient condition. However, this application of embedded inference engine focused on limp detection and gait analysis only.

While contribution of early health monitoring systems was essential to advances in technical areas as communication models, time synchronisation or energy management, so current works focus rather on integration aspects and new classification algorithms. Therefore, results of those very first research should now be re-evaluated in the light of new requirements and new specific applications. As a result while designing the proposed system a decision have been made to build upon successful solutions from prior systems and formulate new operational logic, to match the requirements defined in sections 4.2.2-4.

Proposed system design, presented in the following sections, reuse and redesign the early principles of WBAN/WPAN and telemedical systems, embedding an inference engine and dynamic sensing model to perform integrated and intelligent vital signs monitoring.

4.3.3 Ubiquities health monitoring system outline

The system architecture, illustrated in Figure 4.5, is based on Otto's well established three tier architecture. It spans between WBAN/WPAN networks, made of individual medical sensors, and Wireless Area Network (WAN) connection to the remote site. WBAN tier comprises a number of sensor nodes, each capable of sensing, sampling, processing, and communicating physiological signals. WAN tier encompass external network of interconnected medical services which could be medical professionals, emergency caregivers, physiotherapists, carers or next of kin. The middleware that links these tiers together is deployed as PDA/smartphone device that interface WBAN nodes locally and communicates with parties at the top tier. It provides an integrated sensor nodes management, data aggregation, real-time data processing and transmission.

Main architectural difference, when comparing to Otto's et al. (Otto, Milenkovic, Sanders *et al.* 2006) design, lays in workload, which in terms of data processing, network management and inference algorithms, has been moved from a centralised medical server to the personal assisting device. By emphasising user rather than a medical server through decentralised data processing we aim to reduce the transmission overload and high cost of service maintenance, while at the same time improve the quality and accuracy of obtained results. In our system it is the responsibility of personal server to collected and process data, recognise patterns, detect any anomalies and take accurate decisions. This allows us to eliminate centralised medical server from this architecture and use available external services instead. For instance, there is no need for a patient's physician to access patient's medical records to ensure that he/she is within expected range for each vital sign, whether patient is responding to a given treatment or that the patient has been performing the given exercise. As far as any of the above events occur, the personal server will automatically notify adequate parties. These ensure that both patient as well as medical personnel are feedback with only relevant and up-to-date information.
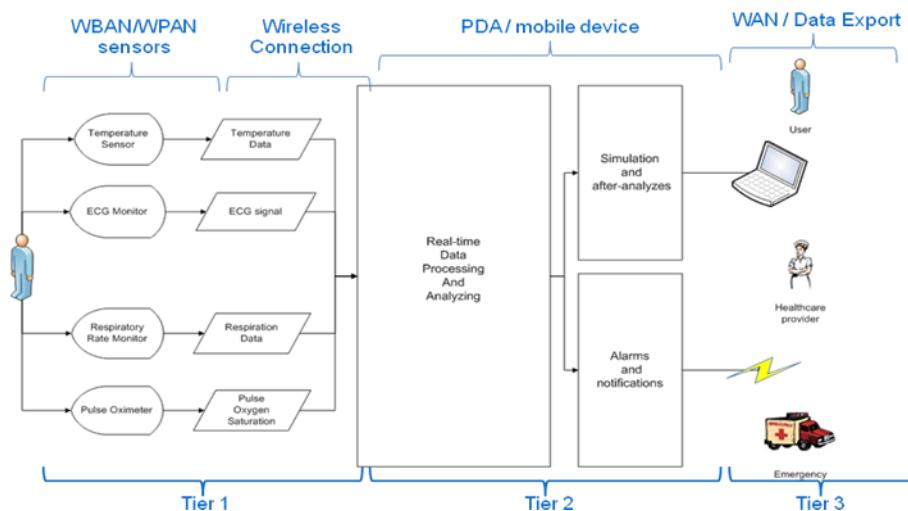
Figure 4.5 System architecture.

4.3.3.1 Tier 1 – Data Acquisition

The first tier of the system is a typical wireless sensor network that consists of distributed sensor devices that are placed on human body. Their main function is to sense in non-intrusive and convenient way different physiological parameters and transfer relevant data to a personal server through wireless network. The fully operational prototype system is made of two separate sensor nodes that are illustrated in Figure 4.6. The first one includes 3-lead electrocardiograph (ECG), respiratory rate sensor (RESP), core body temperature sensor (TEMP) and 3-axes accelerometer (ACCEL). The second node is a pulse oximetry sensor (SpO$_2$). Selected sensors configuration allows us to monitor five different vital signs, which are standards in most medical settings. This are:

- Body Temperature (TEMP)
- Heart Rate (HR) and Electrocardiogram (ECG)
- Blood Pressure (BP)
- Respiratory rate (RESP)
- Pulse oximetry (SpO$_2$)

Sensor nodes, beside their primary function to sample physiological data, perform data filtering and pre-processing. Filtering aims to eliminate undesired noise that comes with a sensed signal. Pre-processing, in turn, depending on the sensed phenomena and the application, converts obtained signals into more meaningful values within assumed scales and perform some initial calculations or validations of data. Such prepared data are then stored locally or transferred wirelessly to the personal server. For the purpose of data transmission it
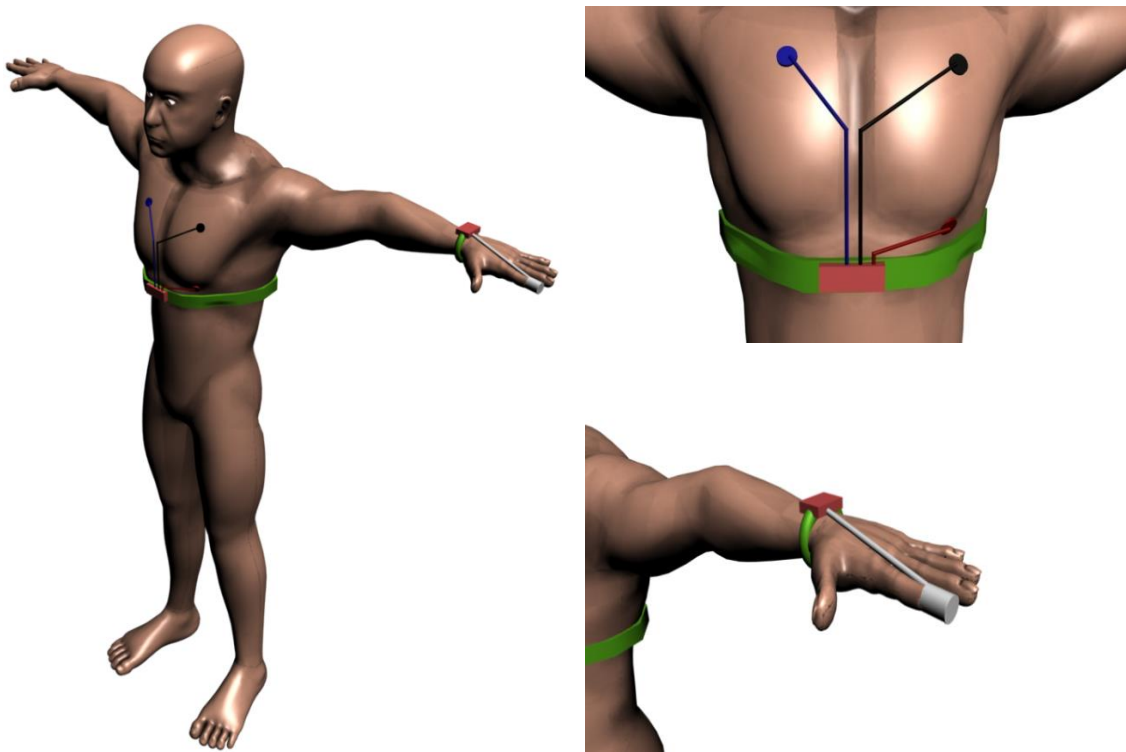
Figure 4.6 Sensor nodes and their placement on human body.

uses specific network data frames. It is also capable of responding to a generic as well as sensor specific queries for data that comes from personal server.

Sensor nodes used in this scenario are deployed on Shimmer platform (Burns, Greene, Mcgrath *et al.* 2010), capable to perform data sampling, filtering and processing using the 8 MHz processor with build in analog-to-digital converter. It is featuring some extension boards that increase the potential to integrate different sensors. It is equipped with a Bluetooth as well as ZigBee (802.15.4) wireless transceivers and flash memory what allows for data transmission and storage. All these features are compacted in a very small form factor. Moreover, by supporting modularised open source TinyOS operating system, Shimmer becomes an easy to implement platform for data acquisition applications.

4.3.3.2   Tier 2 - Personal server

The second tier is the personal server application that runs on PDA/smartphone device. It is the central component of the system, which interfaces the WBAN/WPAN sensor nodes locally and medical professionals or other parties in tier three globally (if such are present in the system). Its main functions are to aggregate and process sensor data.

Data aggregation includes WBAN/WAN network configuration and management. WBAN network configuration encompasses tasks such as nodes discovery, nodes registration (e.g., network protocol, sensors number and types), initialisation (e.g., network frames and buffer allocation, sensors calibration, mode of operation), and setup of a security policy (e.g. devices pairing, packages encryption, checksum). WBAN network management, in turn, is responsible for tasks such as communication channel sharing, time synchronisation, data transmission, and data encryption. Finally, data aggregation element coordinates WAN services. It maintains the communication channel with third parties and when such is needed, establishes a secure connection and sends reports or data that can be integrated into the user's medical record. Data aggregation along with sensor nodes is the core elements of the platform that meets the requirements for ubiquities data sensing and makes real-time data processing feasible.

Data processing focuses purely on classification and decision algorithms that determine the user's state and well-being status, providing feedback to the user or a relevant third party either upon request or in any anomaly or life threatening event. It is the most crucial element of our system that brings the idea of intelligent, ubiquities monitoring into life. By taking advantage of recent improvements in processing power and battery life of modern mobile devices, it is planned to implement/port an inference engine for intelligent data processing, what is one of this project novelties. The challenge is to optimise the solution, which was previously reserved for PC or mainframe machines with high computation power, to fit in small portable systems with restrained resources.

Primary choice for this component is to implement an artificial neural networks engine, however any other inference engine or even hybrid solutions might be used at the later stage of this project depending how well it will perform intended classification. For now such initial

selection was taken due to the fact that ANN can be used to infer a function from observations and also to use it. This is particularly useful in applications, where the complexity of the data or task makes the design of such a function by hand impractical. Health and medical diagnosis are one of those applications, where pattern and sequence recognition, novelty detection and sequential decision making are crucial parts and artificial neural networks were extremely useful in a past (Tu 1996, Saalasti 2003).

4.3.3.3 Tier 3 - WAN/Data distribution

The last tier of this framework encompasses external network of interconnected medical services. This can be medical professionals, emergency caregivers, physiotherapists, researchers etc. It provides data/information exchange between user and medical personnel. This is especially important in case of any life threatening event or anomaly detection, which requires medical consultation.

By including this tier in our system design we do not intend that medical professional or caregiver have to have a constant access to the monitored data while the user is performing continuous every day monitoring. What differs our solution from others is the assumption to limit the interaction between user and third parties to minimum. This means, that the medical professionals service will be used ad-hoc, only when anomaly or emergency is detected by the system. Otherwise, when all parameters remain within their ranges the system will operate as an autonomous personal health assistant. Such approach, significantly reduces the usage of mobile network bandwidth, when comparing to the Otto's (2006) design, and focus on ad-hoc WAN communication channels such as SMS, online services or emergency calls first.

However, in case of medical diagnoses or trials, the continuous form of user – medical personnel communication should be used. This includes remote access through web based graphical user interface or real-time data transmission to the remote location for medically supervised monitoring. This form should give a continuous, real-time insight into user's health status.

The data gathered requires encryption of all sensitive information related to the personal health, addressing information reliability and patient privacy issues, required by law. This also involves making data anonymous as well as control access authorisation and authentication, while transferring data to medical databases. It involves also an issue of service provision due to wireless network availability.

However third tier is included in general system architecture it won't be fully implemented and deployed within the scope of this project, but yet will let to extend the system in the future. The minimum requirements for this project are notifications and alarms. These services will be considered as third tier in remaining chapters.

4.4 Software architecture

4.4.1 Introduction

This section focuses on detailed software architecture of the proposed system. It describes software modules running on SHIMMER (Burns, Greene, Mcgrath *et al.* 2010) sensor nodes and PDA/smartphone device, devoting considerable amount of time to short-range wireless network communication. The main functionality of the modules, described in this section, includes: real time signal sampling and on-sensor pre-processing, WBAN communication, data acquisition and aggregation, data sources control and user-system communication. The only component description, not included in this section, is the data analysing module. Despite of being an integral part of our system, it is presented here as a "black box" with certain inputs and outputs, due to strong dependence on application domain. Its detailed specification can be found later on in chapter dedicated to real time intelligent data processing.

According to this the following sections are organized as follows. First section outlines general components design and discusses their role. Second section details the software architecture of sensor nodes, focusing on components design patterns and interfaces they provide. Finally, the third and last section discusses personal server framework design.

4.4.2 Software architecture overview

The software architecture of our prototype system is illustrated in the block diagram in figure 4.7. It encompasses sensor node applications implemented in nesC (Philip Levis 2006), and personal server application implemented in J2ME (Java 2 Mobile Edition). The first one runs on SHIMMER Platform under TinyOS (Gay, Levis and Culler 2007) operating system. The second one runs on phoneME Java Virtual Machine (Phoneme n.d.) under Windows Mobile 6.5 operating system. The backbone of the system is the Bluetooth (Bluetooth Sig 2007) wireless communication technology, which forms the Wireless Body Area Network and spreads between sensor nodes and personal server.

Since our focus, while designing the software has been on encapsulation of similar functionality into modules, each application consists of number of software components. Components are responsible for specific tasks and in most cases, are standalone elements that facilitate data flow and offer their services to each other through easy to use interfaces. Such services encapsulation improves modules usability, reliability and extension capabilities while at the same time helps to quickly detect and correct any error or faulty operation within the module.
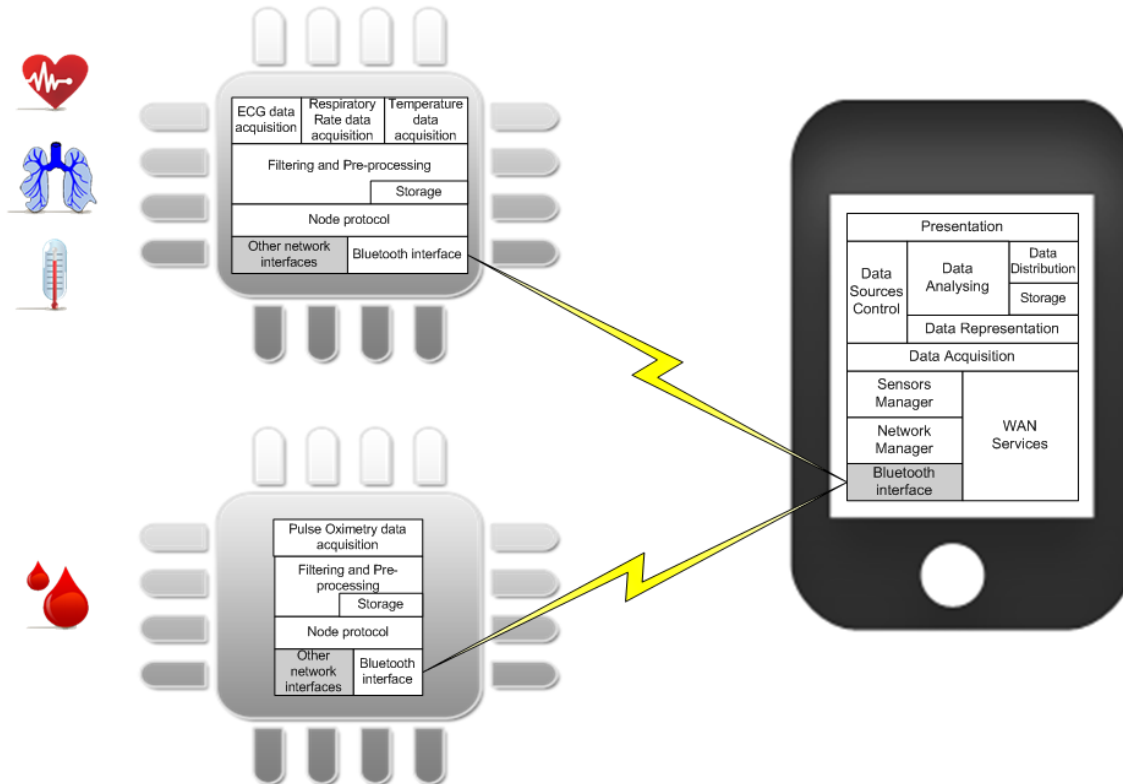
Figure 4.7 The block diagram of system software components.

4.4.2.1 Sensor node components

In this spirit sensor nodes, which are pivotal elements of our framework, consist of a number of specialised software components, which include:

- data sampling component,
- data filtering component,
- data pre-processing component,
- local storage component,
- and network protocol component that offers various data transmissions using Bluetooth or 802.15.4 radio component.

The exemplary use case scenario presents as follows. The data acquisition component samples physiological signals that are present on sensors' input channels of analog-to-digital converter. The sampling process depends on monitored phenomena, signal frequency and associated noise. When quantised signal is obtained, it undergoes a digital filtering in order to eliminate unwanted frequencies. Filtering component offers a selection of signal specific filters. Further conversion of noise-free signal into more meaningful values within signal specific scales, performs at data pre-processing component. Finally, storage component stores obtained date in local flash memory or network protocol component, transmits it wirelessly to the personal server for further processing.

4.4.2.2   Personal server components

Following modular design also personal server framework consists of a number of specialised software packages. These are grouped into two layers: data aggregation and data processing layer.

Data aggregation layer consist of:

- network interface package,
- network abstraction model made of network manager and sensor manager packages,
- WAN services package,
- and network control package.

Data processing layer, in turn, consist of:

- data representation package,
- data sources control package,
- data analysing package,
- storage package,
- data distribution package,
- and GUI presentation package.

Data aggregation layer is mainly responsible for sensor data fusion as well as WBAN and WAN network configuration and management. It provides an abstraction and virtualisation of nodes, sensors and connections, through WBAN network manager, sensor manager and WAN services coordinator modules. Their primary function is to keep register of every single data source and manage them accordingly, including discovery, registration, configuration and initialization. They use various network protocol modules, implemented on top of Bluetooth, Wi-Fi or other radio stack interfaces. Once the network connections are set up, network control package manages network utilization, taking care of channel sharing, time synchronization, data transmission or data encryption. As a result, data aggregation layer provides the complete collection of real-time, pre-processed and cleaned sensor data that is ready for processing.

Data processing layer entirely depends on data aggregation layer. It focuses on real-time classification and decision algorithms, which apply to collection of data supplied by sensor nodes. Data processing base on inference engine deployed as data analysing package. It use data representation package in order to obtain higher level semantic data or indexes used for analyses. Data processing is capable to take control over data acquisition through data sources control package, in order to let the algorithms decide what data is needed and when. This allows us to introduce the dynamic sensors model that utilises only these sensor channels that are necessary at this time for accurate system operation. Data sources control package is a decision making tool for data sources management. Other tools that decision making mechanism can use are: graphical user interface, data distribution mechanism, which posts alarms and notification remotely to third parties, and file system to keep monitoring logs.

4.4.3 Sensor node software

nesC programs consist of components connected, or using a nesC jargon wired, via named **interfaces** that they provide or use. There are two types of components in nesC: **modules**, which provide the implementations of one or more interfaces implemented in C language and **configurations**, which are used to assemble other components together, connecting interfaces used by components to interfaces provided by others.

Components define two scopes: one for their specification, which contains the names of their interfaces, and a second scope for their implementation. The **provided** interfaces represent the functionality that the component provides to its user in its specification; the **used** interfaces represent the functionality the component needs, in order to perform its job in its implementation.

Interfaces are bidirectional. They specify a set of commands and events. **Commands** are functions to be implemented by the interface's provider, while **events** are functions to be implemented by the interface's user. For a component to call the commands in an interface, it must implement the events of that interface. A single component may use or provide multiple interfaces and multiple instances of the same interface.

In the spirit of the basic concepts and syntax of the nesC component model presented above, we have developed prototype sensor node software, which include six components that differ in their structure and functionality. Figure 4.8 graphically depicts the assembly of them connected via their interfaces.
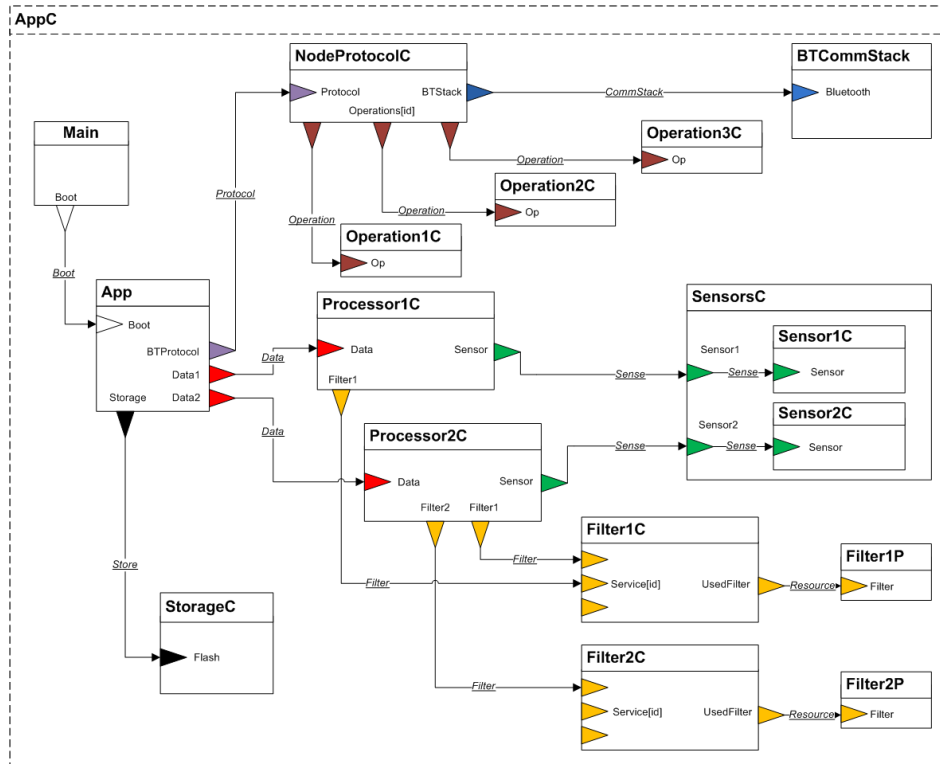
Figure 4.8 The sensor middleware software components assembly.

All application components exist under top level AppC configuration that wires all functional components together. These functional components and their interfaces are:

- `SensorC` and interface `Sense`, are responsible for data sampling on analog-to-digital converter's inputs.

- `Filter#C`, where # depends on the number of filters implemented as `FilterP` modules, that provide `Filter` interface. It is responsible for signal filtering.

- `Processor#C`, where # depends on the number of separate data being monitored, and Data interface. It is responsible for data pre-processing.

- `StorageC` and interface `Store`, responsible for local data storage on flash memory.

- `BTCommStackC` and interface `CommStack`, which implements network interface responsible for data transmission, in this case using Bluetooth radio.

- `NodeProtocolC` that provides `Protocol` interface as well as its subsequent `Operation#C` components, where # depends on the number of operations which provide the proprietary interface `Operation`. They are responsible for performing an externally customizable set of operations in response to the input from the network interface or call from App module.

- and `App` component that merges all components into one application under `AppC` configuration.

According to Han et al. (2005), TinyOS is the current state of the art in sensor node operating systems, where reusable components implement common services, but each node runs a single statically-linked system image, making it hard to run multiple applications or incrementally update applications. Proposed here middleware simplifies code reusability, application flexibility and customization of components with the traditional TinyOS development and deployment model, alike in the model used by SOS operating system proposed by Han et al. (2005). However, there is a significant difference between these two models. Model proposed here is the extension of the TinyOS operating system, while SOS is a completely new operating system for mote-class sensor nodes, which as authors claim takes a more dynamic point on the design spectrum. Proposed here model answers therefore limitations of TinyOS with clearly defined software engineering patterns what enables flexibility, reusability and customisability of WSN applications developed using TinyOS.

The following sections describe each component in details, discussing interfaces they use, their design patterns and interactions between components.

4.4.3.1   Data sampling component

Overview and design pattern motivation

Data sampling component performs continuous signal sampling on up to eight 12-bit ADC sensor channels. Each sensor channel is accessed through provided `Sense` interface wired to and exposed to external components by the `SensorsC` facade component. It provides a unified access point to a set of inter-related sensor services that `Sensor#C` components implement.

A Facade pattern used to implement data sampling component is a configuration component that defines a coherent abstraction boundary by exporting the interfaces of several underlying components. In this case, the components are sensors deployed on sensor board that are used by the application. The use of Facade design pattern allows us to group similar functionality together but still present the interfaces separately. A user can wire to only those sensors it needs, being certain that everything underneath is composed correctly. In case of a data sampling, where the operations are split-phase, it help us also to limit the number of events the component, that wants to use `SensorC` component, have to handle. Despite of being connected to Facade configuration, still all sub-components can be easily decomposed into separate components without introducing internal interfaces. These would be necessary if putting all of the operations for all sensors in a single, shared interface. Finally, Facade pattern allows us to implement complex abstractions of sensors in smaller distinct parts. On one hand side, it simplifies the code, and on the other side, groups the same functionalities under single configuration, what facilitates trouble shooting and clarify design.

`Sense` interface

Interface type `Sense`, responsible for connection between data sampling module and its users, represents a typical split-phase execution model presented in figure 4.9.
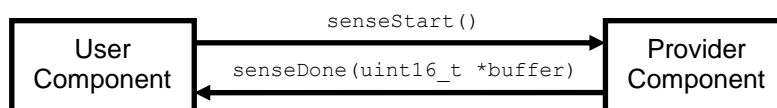
Figure 4.9 Split-phase execution model.

Interface provider, in this model, implements commands that can be called by the interface user. `Sense` interface implements two core commands: `senseStart()`, which request sensor resources and initiate the sampling on particular channel(s) and `senseStop()` that release resources and stops the sensing process. The user of this interface must in turn implement the event `senseDone(uint16_t *buffer),` which the provider signals, when the sensors read completes. The value(s) read are enclosed in the event parameter and can be processed accordingly by `Processor#C` component, where # refers to the corresponding `Sensor#C` component.

Analog-to-Digital Converter interface

The data sampling component is entirely hardware dependent. Each instance of `Sensor#C` component, where # refers to sensor number, uses MSP430 microcontroller specific `Msp430Adc12SingleChannel` or `Msp430Adc12MultiChannel` interfaces, which are part of ADC12 TinyOS distribution library. `Msp430Adc12SingleChannel` interface is used to perform one or more ADC conversions on a single channel with a specific sampling frequency. This interfaces can be used, depending on the application and type of sensor, to sample a single ADC channel once or repeatedly (one event is signalled per conversion result) or perform multiple conversions for a single channel once or repeatedly (one event is signalled per multiple conversion results). If sample should perform on many ADC channels simultaneously, with single command, `Msp430Adc12MultiChannel` interface is used instead. `Msp430Adc12MultiChannel`, is used to sample a group of up to 8 different ADC channels. They provide a way to access the ADC12 on the level of HAL. HAL (Hardware Abstraction Layer) is a middle layer of the Hardware Abstraction Architecture (HAA) of Tiny OS 2.0 (Philip Levis 2006). These lower level, platform dependent interfaces are used due to their better efficiency comparing to the Tiny OS platform independent interfaces, when it comes to high frequency data sampling. Detailed guide into Analog-to-Digital I/O PINs, their configuration and sampling process can be found in Section A3.1 in Appendix 3.

4.4.3.2 Data filtering component
Overview and design pattern motivation

Digital data filtering component removes unwanted noise from the signal. The core system design provides three basic filters: low-pass, band-pass, and high-pass filters. The selection of this filters as a core base for the system was imposed by their wide use in monitoring of different medical phenomena. However, other filters can be implemented assuming their design base on service instance pattern specified below.

The Service Instance pattern was used in design of `Fitler#C` component in order to allow multiple users to have separate instances of a particular service and the instances to be able to collaborate efficiently. It is the basic mechanism for virtualising services that allocates state for each instance and coordinates underlying resources based on all of the instances. The instances share code and can access other's state. A `Filter#C` component, following the Service Instance pattern, provides its services in a parameterised interface. Each user component wires to a unique instance of the interface using `unique` construction. The underlying component receives the unique identity of each client in each command. This is used then to index into a state array in order to retrieve the instance state. The component can determine at the compile time how many instances exist using the `uniqueCount` function and dimension of state array accordingly.

The use of Service Instance pattern in this case allow us to provide multiple instances of a particular filter service, which share the same code, however it does not know how many instances will be needed until the compile time. Moreover, each service appears to its user to be independent of the others in the same way as in traditional object-oriented programming when instantiating an object representing the service. Service Instance pattern allows controlling state allocation, so the amount of RAM used is scaled to exactly the number of instances needed. It conserves the memory, while preventing the run-time failures due to many requests exhausting resources. Finally it allows us to rapidly extend the system by specifying and implementing new filtering algorithm and wire its implementation there, where filter is needed.

`Filter` interface

Interface type `Filter` represents filtering component functionality. Its provider (i.e. `FilterP`) must implement three core functions: `init()`, `reset()`, and `filter(uint16_t *buffer)`. The `init(double cutOff1, double cutOff2, uint16_t numSamples)` method initialize the `FilterP` component to perform filtering on `numSamples` of data from ADC with cut off frequencies at `cutOff1` and `cutOff2`. By calling `init()` any previous configuration will be overwritten. If `SUCCESS` is returned the filtering module is configured correctly and call to `filter(uint16_t *buffer)` will start filtering process. The `reset()` function reset the filter to its initial state. If called when filtering process is on, it interrupts its operation. If call to reset() returns `SUCCESS`, it means that the filter was reset to its initial configuration successfully and either `filter(uint16_t *buffer)` can be called to start the new filtering process or `init()` can be called to reconfigure the filter. Finally when all configuration were successful the call to `filter(uint16_t *buffer)` starts filtering on data buffer passed in parameter. It uses the configuration as specified by the last call to init configuration command. Finally, the user of `Filter` interface must implement the event `filterDone(uint16_t *buffer)` which the `FilterP` signals, when the filtering of all `numSamples` completes. The results of filtering process are passed in the event parameter.

Filter#C component

`Filter#C` component wires the `Filter` interface to the `Filter#P` component implementation. `Filter#P` is the bottom provider of interface type `Filter` that adds filter specific functionality, in this case one of low-pass, band-pass, high-pass filter functions, to the interface.

`Fitler#C` wires these component implementation resources using a very powerful nesC design paradigm, namely: parameterised interface and the `unique` construction. Parameterised interface is an interface array, where array indices are wiring parameters. This construction allows us to wire a number of interfaces in the single line of code, specifying only the indices of

interfaces that should be wired together. However, in cases of using filtering component in many data processing components, the use of a single element of a parameterised interface does not require us to know to which component it is actually wired as long as we are sure that no other processes use it. This functionally is supported by the nesC'c unique construction, which is used by the `Filter#C` configuration, an instance provider for the external filter services users. All uses of `unique("string")` with the same constant argument "`string`" return different values, starting at 0. Therefore we are able to wire a number of instances of some object, where the number of these instances is known at the compile time and can be returned using `uniqueCount("string")` function in the application. Using these paradigms we are able to overcome the limitation of nesC 1.1 language, which could not have multiple copies of components, what required sharing state across objects or state copying, what used additional RAM.

4.4.3.3   Data pre-processing component

Overview and design pattern motivation

Data pre-processing component performs digital signal pre-processing. It involves signal filtering and conversion of such noise-free signal into more meaningful values within signal specific scales. It performs also an initial validation of signal against some threshold values.

The Decorator pattern was used in design of `Processor#C` component in order to enhance capabilities and functionality of `SensorsC` and `Filter#C` components without modifying their implementation. Through Decorator we are able to apply this extension to any component that provides the interface. The operation of Decorator pattern can be compared to a traditional object–oriented inheritance, which defines the relationship at compile time through a class hierarchy.

A `Processor#C` component is a module that provides and uses `Sense` and `Filter` interfaces. The provided interface type `Data` adds functionality on top of this two used interfaces. It takes buffer of sampled values obtained form Sense interface, performs a set of filtering and data conversion activities on it and returns those values that conform to some predefined data validity rules as an event parameter.

Using Decorator pattern can have further benefit for our design and meet the requirements of sensors nodes for rapid extension and adaptation to new applications. In addition to augmenting existing interface, Decorator allows us to introduce also new interfaces. These interfaces can provide alternative abstraction, not only in case of behaviour but also in case of synchronous or asynchronous (split-phase) operation model. Finally by separating the added functionality into `Processor#C` component, allows us to apply it to any implementation. By simply replacing the `Processor#C` component on top of `SensorsC` and `Filter#C` components abstraction can easily and without any inconsistency with bottom layers, meet new

application requirements. It allows us to provide several variants of a component without having to implement each possible combination separately.

`Data` interface

Interface type `Data` represents pre-processing component functionality based on a split-phase execution model. Its provider must implement `init(uint16_t numSamples)` function to control sensor and filters interfaces. In order to trigger data sampling process to begin or stop the provider implements `dataStart()` and `dataStop()` commands. It must implement also function `setTresholds(uint8_t cop1, uint16_t t1, uint8_t op, uint8_t cop2, uint16_t t2)` to set threshold values for each sensor channel. These values are used to trigger or suppress `dataReady()` events of sensor data when the predicate condition is or is not met. The predicate condition has the following form:

$$(data \; \textbf{cop1} \; T1) \; \textbf{op} \; (data \; \textbf{cop2} \; T2) \tag{8}$$

where $data$ is the output of sensor types, $T1$ and $T2$ are threshold values, $\textbf{cop1}$ and $\textbf{cop2}$ represents a comparison operator such as <, <=, >, >=, = or !=, and $\textbf{op}$ is a logical operator one of AND, OR, or XOR. Design of this functionality follows part of (Shnayder, Chen, Lornicz *et al.* 2005) work. It allows no more than two sub expressions that can be included in the predicate. It sufficiently define the subset of interest and allows the predicate to fit in a single query message. The example of such query for the HR could look like (*HR* < 50) *OR* (*HR* > 200), which would trigger `dataReady()` event, when the patient's heart rate falls below 50 bpm or exceeds 200 bpm. Use of this command allows the system to predefine sets/ranges of data that are of interested for the application. Implementation of such functionality at the sensor node allows us to significantly limit amount of data being transferred to the personal server.

Finally as the data pattern of interest occurs the `dataReady()` event is signalled. It delivers in its `uint16_t *buffer` parameter an array of filtered and pre-processed sensor data values. The `buffer` is initially of size `numSamples`, however the actual number of data being in the buffer may vary and depends on the length of sensed data of interest.

4.4.3.4   Storage component

Overview and design pattern motivation

Data storage component represents the file system abstraction. It provides the local sensor node storage on flash memory card, implementing interfaces for writing and reading files. It handles also such meta data operations as create, rename or delete file(s).

Considering fitting all of the above operations in a single, shared interface would raise many problems. The first one would be the fact that in case of split-phase operations each user component would need to handle a number of events, which are actually not necessary for it. This is due to the fact that if a component wants to use any command of the interface it have to

implement all its events. Therefore considering a component that only writes to the file by appending data to an existing file, all meta operations on file system are irrelevant. Moreover, if a developer decides to change the file system or files structure single interface implementation won't be easily decomposed into the subcomponents that the developer could replace. It would involve the redesign and introduction of new components, what would simply increase complexity and make maintenance more difficult.

The Facade design, similar to the one used to implement `SensorC` component, answers all of the above limitation. It relays on a single configuration, where separate modules implement each of the interfaces, depending on common underlying services such as block size of read/write data. Moreover, it defines a coherent abstraction boundary by exporting the interfaces of several underlying components. At the same time by wiring underlying components, it simplifies dependency resolution.

The StorageC facade configuration uses and provides three separate interfaces `FileRead, FileWrite, FileControl`.

`FileRead, FileWrite, FileControl` interfaces

Interfaces type `FileRead<val_t>` and `FileWrite<val_t>` provide generic interfaces for writing and reading files. Generic interfaces characterise with ability to use types passed to them at creation as a single argument in angle brackets. It defines the type of the data value that it uses. Normally, the `val_t` type is a structured type that represents a complete data record rather than a primitive type. Providers of this interfaces implement `write()` and `read()` methods. As the process of writing and reading involves memory access what is much slower than normal arithmetic CPU operation all methods implemented by this interfaces are split-phase. Therefore, in order to catch and perform some tasks on write/read success or failure confirmation the interface users have to handle events `writeDone()` and `readDone()`.

The provider of interface type `FileControl` implements file system meta operations: `makeFile(), makeDir(), deleteFile(), deleteDir(), openFile(), closeFile()`. As a response to the command call it signals event on its job completion, for instance `fileMade(), dirMade(), fileDeleted(), dirDeleted()`, etc.

4.4.3.5   Node protocol component

Overview and design pattern motivation

Node protocol component provides mechanism to execute node operations in response to environmental input such as sensor readings or network packets. Sensor nodes are able to receive different query messages (packets), invoke corresponding operations and response to the query initiator.

Traditional approach to the commands execution, for instance in object-oriented programming, is to use function pointer or objects, that are dynamically registered as call backs.

Usually, the set of operations is known at the compile time therefore these call backs can be replaced by dispatch table compiled into the executable. Such a dispatch table could be build on `switch` statement, which select an action base on string command to integer mapping. However such approach in very inflexible, because any change to the protocol requires reimplementation of the whole component.

Using Dispatcher pattern of `NodeProtocolC` component, we are able to invokes operations using a parameterised interface, based on data identifier. `NodeProtocolC` is the component on top of `CommStack` interface that dispatches commands. It uses Bluetooth radio `BTStack` interface as a source of already parsed out network queries, that are supplied to it by an asynchronous event `commandRecieved(unsigned char *_cmd, unsigned char *_values, unsigned char *_param)`. The operation identifier in this case is the `cmd` field. The dispatcher is independent of what commands the application handles, or what processing those handlers perform. Adding a new handler operation requires a single wire to `NodeProtocolC`. If an application does not wire a operation handler for a certain type, `NodeProtocolC` defaults to a null operation.

Dispatcher connection

Summarising there are two participating components that provide own interfaces. There is a dispatcher that invokes its parameterised interface based on an integer type, and an operation component that implements desired functionality and wires it to a dispatcher. The following code presents a sample wiring:

```
components Dispatcher, Operation1, Operation2;
Dispatcher.Operations[KEY1] -> Operation1.Op;
Dispatcher.Operations[KEY2] -> Operation2.OP;
```

where in our scenario Dispatcher is `NodeProtocolC` component, and `Operation1` and `Operation2` are command proprietary interfaces that depends on the desired functionality.

The use of Dispatcher pattern allows easily extending and modifying the functionality an application supports, by simple wiring an operation to the dispatcher component. Moreover it allows the elements functionality to be implemented independent and re-used when necessary in any other application, again wiring it to the new application dispatcher. Finally, separation of different functionalities simplifies testing and troubleshooting.

4.4.3.6 Network interface component

Overview and design pattern motivation

Network interface component performs direct radio stack signalling, in this case implemented on top of Bluetooth radio. It is responsible for sending and receiving network packages following the network frames specification for each type of message. It is responsible for radio configuration that allows assigning device friendly name, which will be visible to all

sensor node subscribers, configure device class and name of the protocol it uses, configure authentication by assigning new PIN number used in paring process or turn on data encryption.

This functionality is encapsulated in `BTCommStackC` component which in its structure base on Adapter pattern. It converts the interface of a component, in this case `Bluetooth`, `Init` and `StdControl` into a single interface type `CommStack,` which provides abstraction of the radio commands, configured to handle our network specific frame types and signalling protocol. It includes also some user LED communication patterns, which allow the user to control the status of the node: whether is on or off, in date transmission mode or data sampling etc. Implementing `BTCommStackC` adaptor some of the network frame's parsing mechanisms were implemented on top of standard Bluetooth stack, which simplifies access to network resources through higher level `CommStack` interface.

`CommStack` interface

Interface type `CommStack` represents network interface abstraction functionality for sending and receiving network packages base on a split-phase execution model as well as configuring network radio through set of predefined commands.

Its provider must implement functions to configure and control data transmission. This functions include: `boot(unsigned char *name, unsigned char *pin)` command, which initialize and configure device by assigning its name and desired PIN number, control commands `start()` and `stop()` to start or stop Bluetooth radio module in order to decrease energy consumption by the node, or `send(unsigned char *_sendPackage)` command that sends data package to its receiver.

The user of this interface must implement a number of asynchronous events that are signalled, when particular action is performed. This include `connectionMade(uint8_t status)` event, which is signalled when connection with the sensors node is established and `connectionClosed(uint8_t reason),` which is signalled when connection is terminated. Events, which must be implemented by the interface user, include also `commandRecieved(unsigned char *_cmd,unsigned char *_values, unsigned char *_param)` event. It is signalled when the network query frame is received and parsed out correctly to the command, values, and parameters. This event is used by the `NetworkProtocolC` dispatcher to handle the network command by mapping it to the corresponding node operation.

4.4.4 Personal server software

Personal server application consists of two main layers. These are data acquisition and data analysing layers. Data aggregation layer is mainly responsible for sensor data fusion as well as WBAN and WAN network configuration and management. Data analysing layer, in turn,

focus purely on classification and decision algorithms that determine the user's state and well-being status, providing feedback to the user or a relevant third party either upon request or in any anomaly or life threatening event. Other personal server packages, described below, include graphical user interface, data sources control, data representation, data distribution and storage.

4.4.4.1   Data aggregation layer

4.4.4.1.1   Network package

WBAN network module provides an abstraction of sensor nodes. Its main functionality is to remotely configure and manage a set of physical sensor nodes. It is able to discover and recognise on-board sensors, connect and communicate with them as well as keep register of every single sensor node in the network.

The classes and the interfaces of the `org.j2me4wsn.` `network` package, which implements this functionality, are presented in figure 4.10.
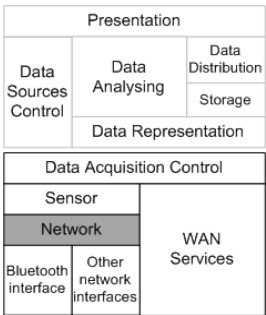
Figure 4.10 Network package.

Nodes discovery and registration

An application can search for `CommStack` based sensor nodes in range using `NetworkManager`. It acts as sensor nodes finder, nodes and protocols register as well as nodes availability monitor. `findNodes()` method of `NetworkManager` returns an array of matching `NodeInfo` objects. `NodeInfo` object contains the information of node properties such as name, MAC address, communication stack and connection URL. Once the sensor node is discovered, the list of available sensors can be retrieved with the `getSensorInfo()` method.

After selecting the desired sensor node(s), the registration process begins. Information contained in `NodeInfo` objects is used to create the `Node` objects. Such objects are registered with NetworkManager by adding them to the vector of registered sensor nodes using `addNode()` method. Additionally, registration process allows us to serialize `Node` information for future use. It includes serialization of `Node`, `NodeInfo`, `NodeConnection`, and `Frame` objects, which are used for establishing a connection and data transmission.
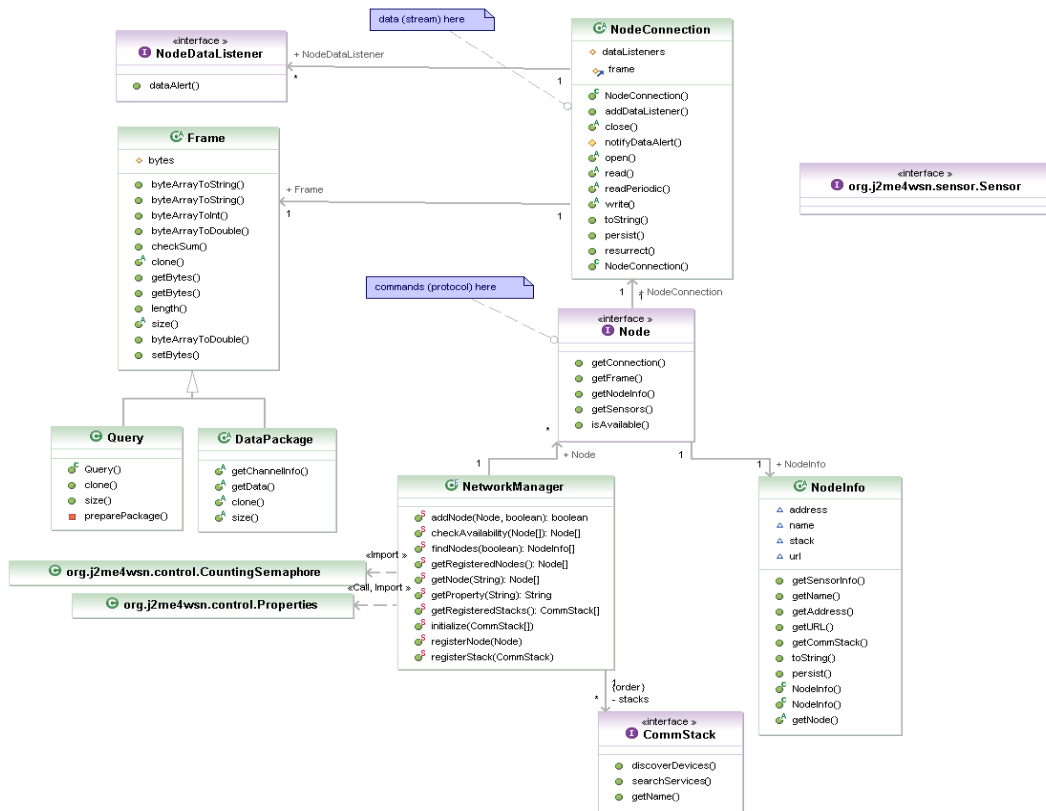
Figure 4.11 Class diagram of `org.j2me4wsn.network` package.

Nodes abstraction

The `Node` interface is an abstraction of an actual sensor node. It provides a set of methods that the application can use to manage each single node. The most important one, `getConnection()` returns a `NodeConnection` instance that is used to connect to the sensor node. Other methods include: `getFrame()` that fetch and return the vendor specific `Frame` object used as data carrier, `isAvalable()` that checks node availability, `getNodeInfo()` that updates `NodeInfo` information and `getSensors()` that returns an array of `SensorInfo` objects, representing available sensors. The last method returns sensor information according to the specified inquiry access code. If accessCode is `true` than performs a remote search at the physical node level, otherwise returns pre-known sensors only. If `null` is returned with accessCode `true` then node doesn't have any sensor deployed and is used as a network repeater only.

Nodes Connection

When the registration process finish the sensor URL contained in `NodeInfo` object is used to establish connection to the node via the `NodeConnection` class. The `open()` method activates input and output connection to the node. An open `NodeConnection` instance can be used to send and receive the data to and from a sensor node. The data can be retrieved either synchronously or asynchronously. Synchronous data retrieval is done by

iterative calls to `read()` method. The asynchronous mode requires applications to register as `NodeDataListener` objects in order to receive `dataAlert()` notifications of collected data. Besides this, `NodeConnection` is also capable of sending data to the network or specific node only. This can only be done asynchronously by calling `write()` method. Finally, when `NodeConnection` is no longer required, call to `close()` method closes this input and output connection and releases any system resources associated with this stream.

According to the above, the `NodeConnection` can enter into four possible states:

1. STATE_OPENED
2. STATE_READING
3. STATE_WRITING
4. STATE_CLOSED

The `NodeConnection` enters the STATE_OPENED state, when `open()` method is invoked. At this point `NodeConnection` can move into one of two different states. When the `NodeDataListener` is registered, the state moves to STATE_READING state and remains in this state until the `NodeDataListener` is removed. When this happens, it moves back to the STATE_OPENED state. Otherwise, if the `write()` method is invoked, the state moves to STATE_WRITING and remains in this state until `read()` method receives the confirmation or end-of-file message from its `Node`. Because the `NodeConnection` can enter into STATE_WRITING also in the different manner, through asynchronous interruption to STATE_READING, `NodeConnection` always moves back to its initial state, where `write()`

Figure 4.12 State diagram of `NodeConnection` class

method was invoked. Therefore, there is no option for `NodeConnection` to move the transitive path from STATE_OPENED through STATE_WRITING to STATE_READING and backwards. This path is forbidden and leads to an exception call. Finally, after the `close()` method is called, the `NodeConnection` is in its final state, STATE_CLOSED. There is no longer any way back to the other states. No data MUST be provided for / send by the application after the connection is closed.

Data transmission

Both synchronous and asynchronous transmission delivers the data/queries encapsulated in `Frame` objects. The `Frame` interface represents abstract data carrier type that contains generic data stored as byte array. The content and function of the `Frame` object can be configured according to specific application or sensor node requirements. There are two default data carriers implementations: `Query` and `DataPackage` frame.

`Query` frame is used to carry network commands that allow the application to remotely control sensor nodes. The default content of `Query` object accommodate the string command, integer values and char parameters that can be send wirelessly to the sensor node. However, custom content can be configured by subclassing `Query` class and overriding the protected `preparePackage()` method. It parses string command according to network command structure. Such packed `Query` frame is converted to its byte representation, following a big endian notation (also known as network byte order) and sent. `Query` specifies whether an application should wait for query confirmation from the sensor node. By default the confirmation is enabled, causing an application to block until the response is received or the `Query` specific timeout occurs. The following figure presents sample Bluetooth `Query` frame format.
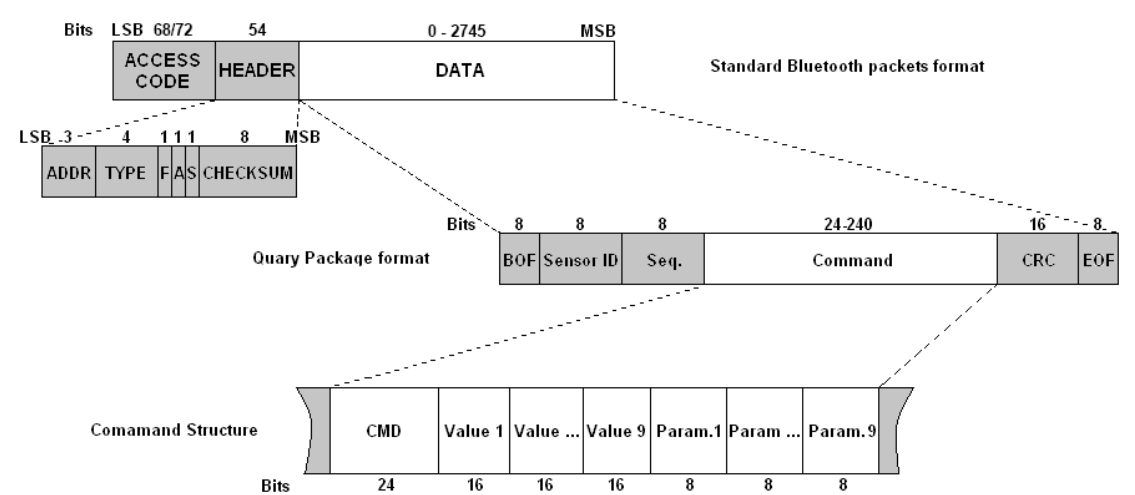
Figure 4.13 The `Query` frame format.

`DataPackage`, in turn, is an abstract class that carries sampled sensor data. Its content can be configured by sub classing `DataPackage` class and overriding the `getData()` method with a sensor node specific frame specification. This method is used to unpack frames, again following big endian notation. Each unpacked sensor measurement has its own channel. Channels represent different dimensions of the measurement. For example, in 3-axis accelerometer, there are three dimensions - axis x, y, and z, which use three different channels. Sensors' data from all the channels is returned simultaneously as a multidimensional array of integer or double values. Several data values may be also buffered inside a `DataPackage` object. The application is able to configure the buffering of the data with a buffer size parameter of `getData()` method. In order to optimize the application performance, `DataPackage` buffering with a larger buffer size is recommended when the high data precision is not required. It allows application to save memory, which in case of mobile device is a limited resource. For instance, if the sampling rate is, 1000 Hz, by setting the size of the buffer to 100, only 10 measurements per second are obtained. This is far easier to handle than 1000 values per second. Moreover, `DataPackage` allows also to retrieve such control data as timestamp, cyclic redundancy checksum or channel info data, as well as sensor node specific control data that can be implemented with the application. The following figure presents sample Bluetooth `DataPackage` frame format.

`Query` and `DataPackage` objects are sensor node dependent objects; therefore inclusion of any new sensor into this framework requires assigning sensor specific frames and wiring them with the specific `Node` object. The framework supports multiple `Query` and `DataPackage` objects simultaneously.
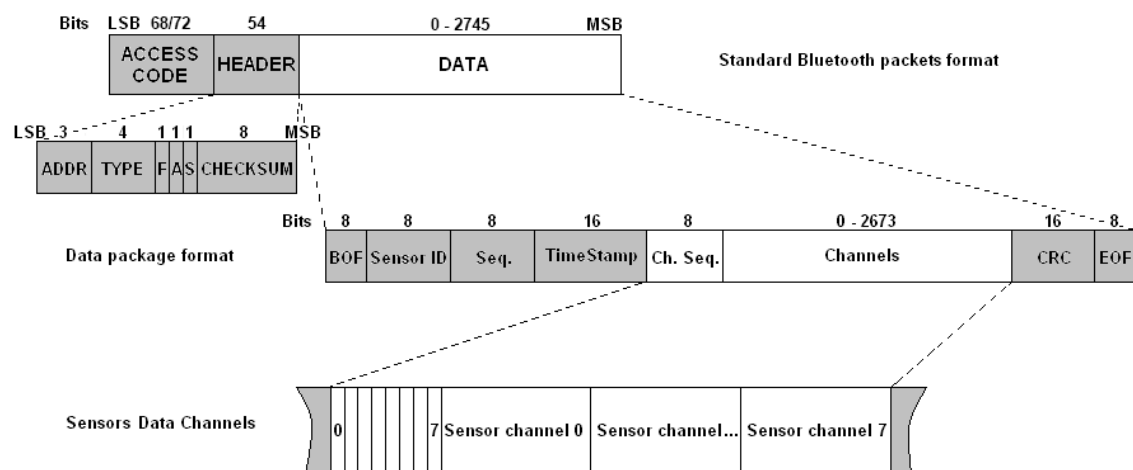
Figure 4.14 The DataPackage frame format.

4.4.4.1.2 Network interface package

Network interface is the bottom layer block that is responsible for direct node connection and signalling using wireless radio communication. It is strongly sensor node and wireless technology dependent module that implements node specific communication protocol on top of existing short range wireless technologies, in this case Bluetooth stack. Its main function is to discover and search for services of physical Bluetooth featured devices in range, establish connection and perform byte transmission. Its design will be discussed based on custom SHIMMER network protocol example,

Figure 4.15 Network interface package

deployed as `communication.bluetooth` package. The classes of this package are presented in the following figure.
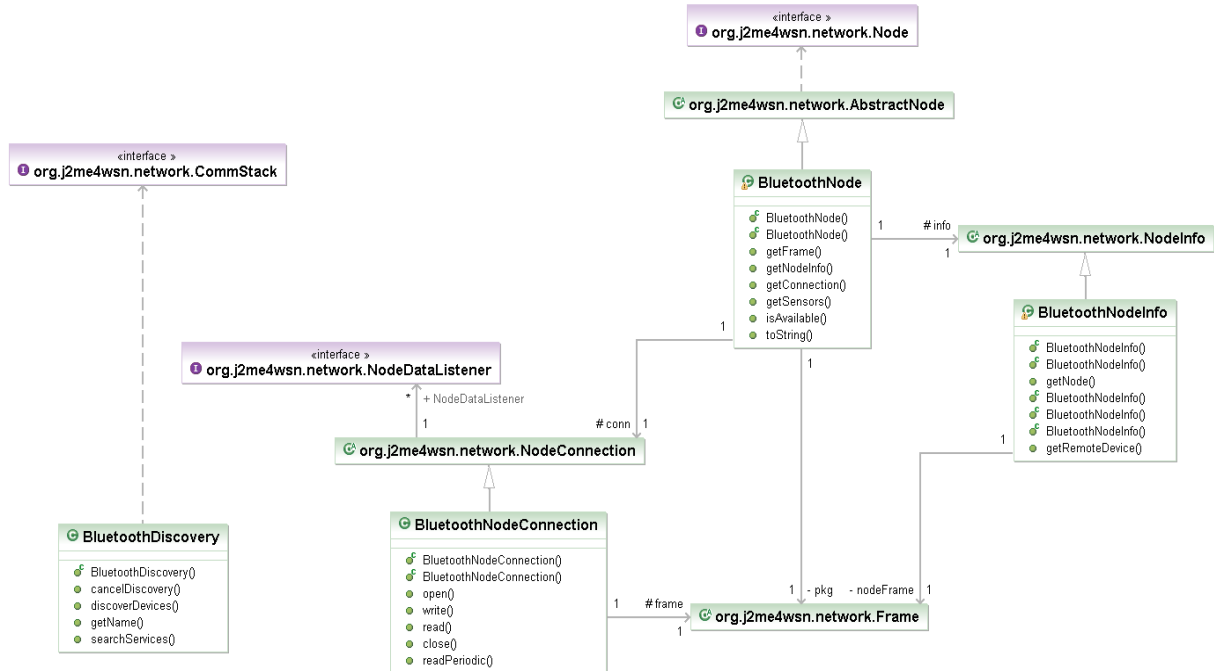
Figure 4.16 The class diagram of `org.j2me4wsn.communication.bluetooth` package.

Porting communication stack

Main element of `org.j2me4wsn.communication.bluetooth` package is the BluetoothStack class, which implements `CommStack` interface. It is a link between the `org.j2me4wsn.network` package and short-range wireless stack, in this case Bluecove (Bluecove Team 2008) implementation of JSR-82 API. BluetoothStack object provides nodes discovery and services search functionality.

Its `discoverDevices()` method places the device into devices discover inquiry mode and searches for Bluetooth capable devices in range. Devices that responded to the inquiry are returned as vector of `NodeInfo` objects to the application. The `cancelDiscovery()` method is called to stop the inquiry.

In turn, `searchServices()` method, searches for services on a remote Bluetooth device according to the service universal unique identifier specified as 32-bit UUIDs. The set of UUIDs, that are searched for, can be configured with `uuidSet` parameter of `searchServices()` method. There is a number of standard services being implemented on top of Bluetooth stacks and theirs UUIDs can be found here (Bluetooth Sig 2007). By default service used to communicate with sensor nodes is serial port profile SPP (1101$_{hex}$). This profile emulates serial port on top of the RFCOMM protocol, based on RS-232 standard for serial binary single-ended data and control signals between a DTE (Data Terminal Equipment) and a DCE (Data Circuit-terminating Equipment). As its output `searchServices()` method returns an array of URL addresses of available SPP featured Bluetooth devices. The Augmented Backus-Naur Form (ABNF) syntax is defined for Bluetooth serial port profile following the guidelines of RFC 2234 (Crocker 1997). When the connection to the SPP is opened using `Connector.open()`, the URL must conform to the ABNF syntax specified in Table 4.1 below.

`<spp_url>`	`:= <protocol><colon><slashes><host>(0*3<params>)`
`<protocol>`	`:= "btspp"`
`<colon>`	`:= ":"`
`<slashes>`	`:= "//"`
`<host>`	`:= <address><colon><channel>`
`<address>`	`:= 12*12(<HEXDIG>)`
`<channel>`	`:= %d 1-30;`
`<params>`	`:= <master> / <encrypt> / <authenticate>`
`<master>`	`:= ";master=" <bool>        *)see constraints note below`
`<encrypt>`	`:= ";encrypt=" <bool>        *)see constraints note below`
`<authenticate>`	`:= ";authenticate=" <bool>    *)see constraints note below`
`<bool>`	`:= "true" / "false"`

Table 4.1 The URL ABNF syntax for Bluetooth serial port connection.

The core rules from RFC 2234 that are being referenced are: `%d` for decimal numbers, and `HEXDIG` for hexadecimal digits (0-9, a-f, AF). RFC 2234 specifies the values of literal text strings as being case-insensitive. For example, the rule master in the preceding ABNF allows all of (";MASTER=", ";master=", ";MaStEr=") as legal values. The URL produced from the `spp_url` rules must not contain both the substrings `";authenticate=false"` and `";encrypt=true"`. Additionally, the URL produced must not contain one of the params (`master, encrypt, authenticate`) repeated more than once. These constraints are being specified here, because ABNF does not contain a rule to achieve the desired functionality.

Examples of valid SPP URL for `Connector.open()` present as follows:

- `btspp://0080375A0001:1;authenticate=false;encrypt=false;master=false`
- `btspp://0050671A03B7:14;authenticate=true;encrypt=true;`
- `btspp://00BB001C0111:3;authenticate=false;`

Porting custom node

Porting of any custom sensor node into the framework requires a developer to implement `Node`, `NodeInfo` and `NodeConnection` interfaces to meet specific node protocol requirements. Classes that implement our default custom SHIMMER network protocol are: `BluetoothNode, BluetoothNodeInfo` and `BluetoothNodeConnection`.

Each sensor node is represented in the system by an instance of a node specific implementation of `Node` interface. This interface contains the abstract methods for initializing and configuring the node. The `BluetoothNode` class provides implementation of SHIMMER node specific methods. It extends helper `AbstractNode` class which implements unidirectional one-to-one associations with `NodeInfo` and `NodeConnection` as well as `persist()` and `resurrect()` methods to serialize `BluetoothNode`. Implementation of remaining `Node` interface methods, described in Appendix 3 Table A3.1, is the way to plug a specific sensor node.

The `BluetoothNodeInfo` class provides access to variety of information about the physical Bluetooth sensor node. It provides an implementation of abstract class `NodeInfo` that allows to instantiate custom node object, in this case SHIMMER node. It is capable to store as well as retrieve node specific information that are necessary to establish connection to the node and obtain sensor specific information. The `BluetoothNodeInfo` class use `javax.bluetooth.RemoteDevice` object to refer to stack specific methods such as `getFriendlyName()` or `getBluetoothAddress()` in order to retrieve discovered nodes information. Implementation of `NodeInfo` stack specific constructors, used to instantiate `NodeInfo` objects, and `getNode()` method, described in Appendix 3 Table A3.2, is a way to plug a specific sensor node info into the framework.

The `BluetoothNodeConnection` class provides node specific implementation of `NodeConnection` abstract class which handles data communication between personal server and this specific node. It is responsible for radio signalling according to specific node protocol. It must implement methods to open and close data stream on particular node connection and perform write and read operations in order to retrieve and parse specified frame objects. Implementation of `BluetoothNodeConnection` methods, described in Appendix 3 Table A3.3, is a way to plug a specific sensor node connection into the framework.

4.4.4.1.3 Sensor package

Sensor package, comparing to the network package is more abstract concept, which configures and manages virtualised set of sensors for the entire network. Its main functionality is to fetch sensor data either continuously or base on some set conditions. From the sensor package perspective it doesn't matter, at which node the sensor is physically deployed, what is the URL address or what is the number of sensors per node. This information are obtained from node while registering it with the system and maintained by the network package.
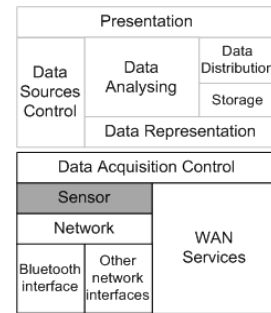
Figure 4.17 Sensor package.

Sensor package instead, perceive sensor as a single, autonomous data acquisition service within any type of network. It maintains information about the physical sensor such as channel it utilise, data type, quantity of data, context of use etc. The appropriate sensor has to be found on node level using network package, or known beforehand, to be able to open the connection using node information from network package. Data can be received once the connection is opened. This simplifies the connection and listening process by hiding whole network specific functionality such as nodes discovery, registration, connection and configuration, when it comes to data processing and data sources management.

The design of the `org.j2me4wsn.sensor`, presented in Figure 4.18, is based on JSR 256 Mobile Sensor API (Nokia Corporation 2006). JSR 256 Mobile Sensor API allows Java ME application developers to fetch data easily and uniformly from sensors. However, in theory the sensor may be connected to the mobile device in different ways, using embedded, short-range wireless, wired, or remote connection represented by the connection type, practically porting of JSR 256 Mobile Sensor Api to be used with any type of Java ME featured mobile device is a complicated task, which often involves native libraries development. Moreover the API does not provide any methods to control a sensor. It only provides methods for receiving the information from a sensor. Control methods are left out for simplicity and the API implementation should take care of this functionality automatically. However, considering that many applications use various number of different sensors hardly ever coming from same vendor with same control protocol, this might be a pitfall. Using JSR256 Mobile Sensors Api we might end up having to implement such protocol, beginning the whole porting process from scratch.

Bearing in mind the requirements that the system must support integration of new sensor nodes from different vendors with different protocols (*F. Req. 11.2*), support different communication technologies for personal server within the wireless body area network (*F. Req. 12*), and finally, allow simple and straightforward configuration (*N. Req. 10*), we decided to make use of Mobile Sensor Api design by redesigning it in such a way to allow operation of sensor package on top of network package. This will allow network package to take care of node and vendor specific protocols, providing a simple way to port new sensor node protocol in Java. `org.j2me4wsn.sensor` in turn will provide easy to use and unified way to manage sensors using the similar interface to JSR 256 Mobile Sensor Api example.
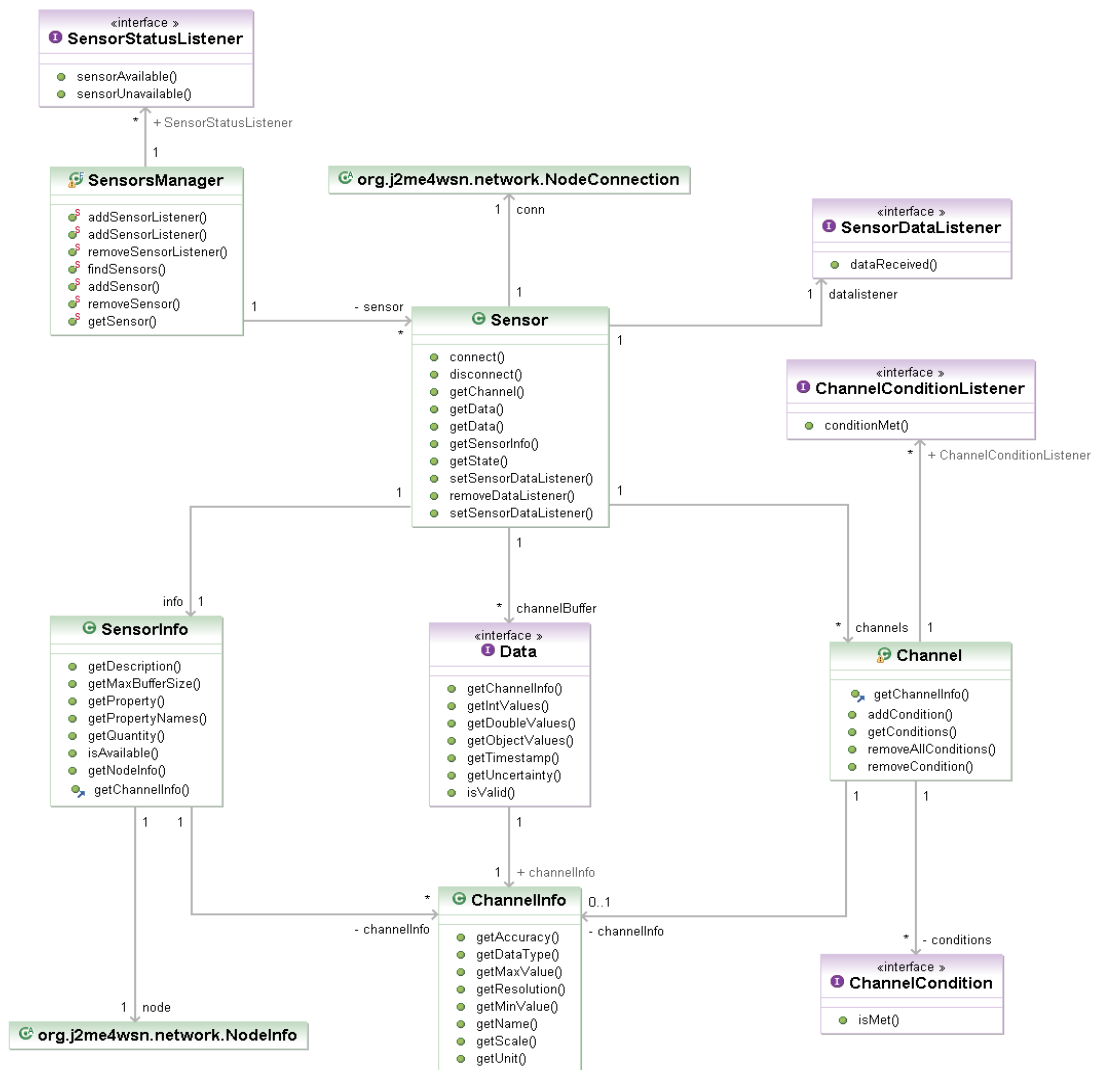
Figure 4.18 The class diagram of `org.j2me4wsn.sensor` package.

Sensor discovery

An application can search for `Sensors` in range using `SensorsManager.` It is used to find sensors and monitor their availability. It provides `findSensors()` method that returns an array of `SensorInfo` objects listing the found sensors. Depending on the `accessCode` parameter it performs remote sensors search at the physical node level using `Node.getSensors()` method that implements node specific protocol, otherwise returns pre-known sensors only.

`SensorsManager` is also responsible for registering and unregistering `SensorStatusListener` objects which monitor availability of the sensor. The listener will receive `sensorAvailable()` or `sensorUnavailable()` notifications. Only one notification for each matching `SensorStatusListener` is sent per change in availability.

Sensor Abstraction

The abstraction of the physical sensor is achieved through number of objects. The main `Sensor` class is an abstraction of an actual sensor. It uses a variety of information about the physical sensor, which is provided by the `SensorInfo` object. It contains information of the sensor as a whole and specifies such information as the quantity, which defines the quantity the sensor is measuring, `Node` object, where sensor is deployed, maximum buffer size for this sensor or the description, the verbal description of the purpose of the sensor.

More specific information of the properties of data that the sensor supply are encapsulated in `ChannelInfo` objects. Single sensor may have many different channels. If the sensor is measuring different properties or dimensions simultaneously, such as ECG sensor is monitoring heart electrical activity on different leads, those values are considered as separate channels. Synonyms for the concept "channel" could be a "stream" "axis" or "degree of freedom". Measurement data from each channel is delivered in a separate `Data` object. The `Data` object contains the information about the channel of its origin and it can be retrieved with the `Data.getChannelInfo()` method. The `ChannelInfo` object has a name and provides such information as data type, data accuracy and resolution, its maximum and minimum values, data unit and scale. This information are used to instantiate the `Channel` object which opens a data stream from the remote sensor applying some data filtering in order to monitor values of interest.

Sensor connection

The `Sensor` class being an abstraction of an actual sensor provides also a connection to the physical sensors through `NodeConnection` object. `Sensor` class provides a set of methods that the application can use to configure, set up and manage each single sensor connection. The connection is established using `connect()` method. It use `SensorInfo` object in order to determine a variety of information about the physical sensor including URL to the node, where the sensor is actually deployed.

When the connection is establishes the `Sensor` object switch to one of the data fetching modes: synchronous or asynchronous. Synchronous mode means that the application is iteratively calling `getData()` method. Asynchronous mode requires the application to implement the `SensorDataListener` interface. When the application implements this interface and registers itself with the `setDataListener()` method, it starts to receive `SensorDataListener.dataReceived()` notifications.

The `Sensor` may provide simultaneous measurements from different channels. `Channel` objects represents data sources and are means of data streams that are buffered in `Data` objects. For the sake of clarity different measurements are stored into separate `Data` objects. The number of channels depends on the nature of the sensor and can be obtained through `getChannel()` method. Finally `disconnect()` method terminates sensor connection, closes

the input stream and releases any system resources associated with this stream. A closed stream cannot perform output operations and cannot be reopened.

Data fetching

Data fetching is done through the `Data` interface that represents data values retrieved from one channel of a sensor. Data from all the channels is returned simultaneously as an array of `Data` objects. It is also a buffer where several data values may be stored. The data type can be integer, double or object. Depending on the data type of the channel, the method used to retrieve the values is different and includes: `getIntValues()`, `getDoubleValues()` and `getObjectValues()`. The information what is the sensor data type can be queried using the method `ChannelInfo.getDataType()`. The `Data` object may also contain the uncertainties, timestamps, and validities of the corresponding data values. The content of the Data depends on the parameters the application specified when starting the data fetching.

When buffering values at `Data` object the buffer size has to be specified. The maximum buffer size of this sensor can be obtained with the method `getMaxBufferSize()`. The data fetching methods will throw an `IllegalArgumentException` if the desired buffer size exceeds defined maximum. In order to improve application performance in case of high frequency data sources the larger buffer should be used. In order to optimize the performance of the application also by omitting the supporting parameters timestamp, uncertainty, or validity, or all of these can help. For more information how to implement data transmission see Network package section.

Data conditions monitoring

An application can monitor fetched data to receive a notification when the data meets the defined condition. This functionality is achieved by `Condition` objects that are used for monitoring the sensor data. They are applied to `Channel` objects. The application defines them by implementing the `Condition` interface. The interface offers the common functionality to check if the condition was met. Each class implementing `Condition` defines specific methods to query the limits and parameters of the condition, as the limits and parameters differ depending on the specific type of condition. Set limits can be asked separately from each `Condition` implementation with special methods, because limits differ in type and number.

When the `Condition` is met, the registered `ConditionListener` receives `conditionMet`() notification. The application has to implement the `ConditionListener` in order to catch the notification and act appropriately in result.

4.4.4.1.4 WAN services

WAN services coordinator provides the integration of the system into wider telemedical framework. Its main responsibility is to maintain wireless connection, send and receive packages and accept incoming remote connections. Their implementation is based on a static `WanManager` object which is responsible for connection status monitoring. It implements the `ServiceStatusListener` which notify when monitored service is available by posting `serviceAvailable()` or `serviceUnavailable()` notifications. Each `Service` object can be in STATE_UP, STATE_TRANSMITTING or STATE_DOWN. Service

Figure 4.19 WAN Services package

provides the input and output stream that are used in order to establish the connection and transmit data in or out of the personal server. The classes and the interfaces, which implement this functionality, are presented in the following Figure 4.20.
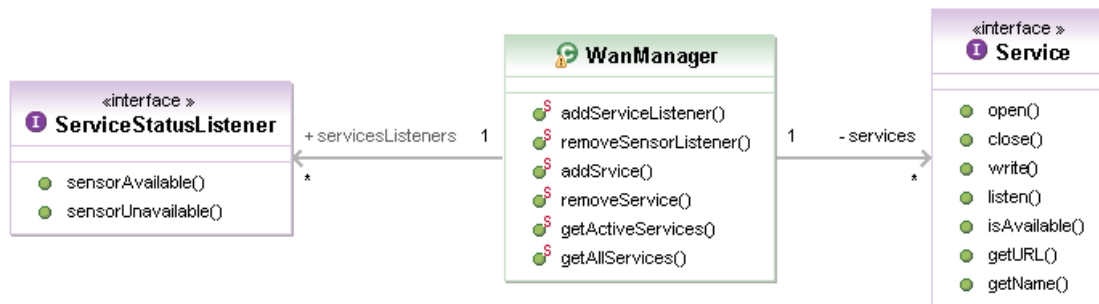
Figure 4.20 The class diagram of `org.j2me4wsn.wan` package.

WAN Services package includes cellular as well as TCP/IP based connections and is capable to handle SMS and voice communication as well as HTTP/HTTPS or SSH protocols. The availability of this service depends on the device. The Figure 4.21 presents the block diagram of WAN communication components.
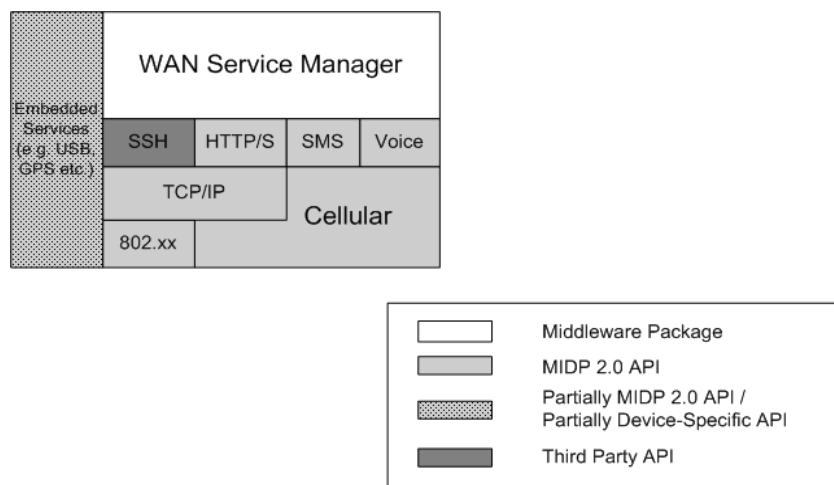
Figure 4.21 WAN service components.

4.4.4.1.5   Data acquisition control package

Data acquisition control package provides network control algorithms. Its main functionality is to control high frequency data acquisition over wireless network to the personal server. It is responsible for synchronisation of many WBAN nodes into a network and controls the network traffic by synchronization of network frames in time. Default network topology is a star network however design of the node abstraction allows to easily implement algorithms, which form others, such as, tree based network topologies. Star network topology spans between a sensor node and the network coordinator, which is the personal server device.
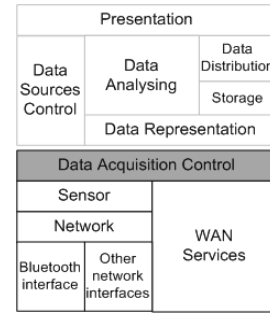
Figure 4.22 Data acquisition control package

Algorithm selected to control the data transmission within such network is a modified version of the Flooding Time Synchronization Protocol (FTSP) developed at Vanderbilt University (Miklos, Kusy, Simon *et al.* 2004). The algorithm divides the network devices on master and slave nodes. The master node is one and constant represented by the personal server. Slave nodes, in turn, are all other nodes registered in the network. Please note that due to this assumption the maximum number of slave nodes is limited by the network interface protocol, in this case Bluetooth, to one piconet which allows one master device to interconnect with up to seven active slave devices. Moreover, creation of scatternets in this protocol, by the slave nodes is not allowed.

Nodes synchronisation is done by the super frame which is divided into timeslots used for data transmissions. Figure 4.23 illustrates principles of super frame operation. Each node uses its corresponding timeslot to transmit sensor data, commands or other specific frames. The first timeslot of each super frame belongs to the network coordinator, which use it for transmitting
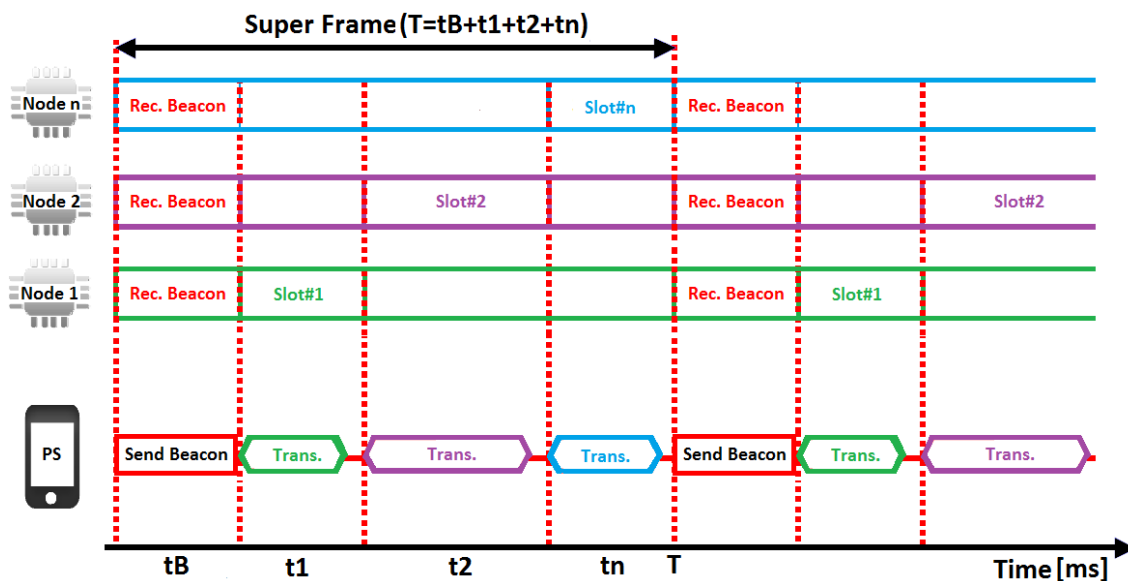
Figure 4.23 The Modified Flooding Time Synchronization Protocol (FTSP).

network configuration periodic beacon frames. The original FTSP protocol use beacon messages to synchronise the start of super frames commands from the personal server. Our modification to the original FTSP protocol includes modifications done by Otto et al.(Otto, Milenkovic, Sanders *et al.* 2006), where beacon frames distribute the global timestamps and our own modification, where beacon carries the command messages to their destination. In such case beacons are used by sensor nodes as a timing reference, to update their local clock, as a synchronization messages as well as commands carrier.

FTSP Protocol modification includes also a timeslot, used to transmit sensor data, which in our algorithm is dynamically allocated to the node that wants to transmit data. The timeslot when the next transmission should happen is notified by the previous package. Also timeslot length is dynamically allocated depending on the super frame length and number of nodes. The dynamic slot allocation aims to efficiently use the available bandwidth, increase the network module sleep time in case of no data for transmission. The dynamic timeslot length, in turn, allows incorporating variable network frame size into the network. It helps to reduce the amount of time in the allocated timeslot, when the node sleeps and use it for such activities as retransmission of some offline data from the node local storage. This allows obtaining some additional information from node repository without any impact on real-time data transmission.

4.4.4.2   Data processing layer

4.4.4.2.1   Data representation package

Data representation package aggregated signal data fetched by `org.j2me4wsn` package into higher level semantic data. It allows to calculate indexes or indirect data indicators used for analyses.

In order to provide a collection of measured physiological data, as higher level semantic data, the package obtain a plain physiological data from the data acquisition module and aggregated data into tuples. The tuples are XML encoded, time indexed, machine-readable well-formed data that follows the grammatical rules specified in XML Schema Definition (XSD) files. Such prepared data, send together for instance to the remote location for medical personnel consultation provides both physiological data as well as meta model. It allows, for instance, for the ad-hoc data transmission to the nearest medical services, where this data can be inspected in case of emergency, without prior knowledge of types or configuration of monitored parameters. All this information is enclosed in the same data pack that data distribution package send over.

Data representation package provides also a stream of data to the data analysing module. It prepares data for inference engine use by aggregating them into engine specific normalised, input sets. Data normalization, refers to the process of identifying and removing the systematic effects, and brings the data from different microarrays onto a common scale. It ensures that data is engine-readable. Data preparation at this stage involves also indirect vital signs calculations in order to use them as additional indicators in data processing.

4.4.4.2.2   Data sources control package

Data sources control package provides and interface to the data processing module capable to take control over data acquisition. This allows us to introduce the dynamic sensors model that utilizes only these sensor channels that are necessary at this time for accurate system operation. Data sources control module is a decision making tool for data sources management.

The inference engine is capable to determine if the diagnosis meet the required level of accuracy or not. If not, for instance, due to insufficient information about the monitored subject, the decision is made as to which sensor from the set of available sensors should be used in next data processing iteration. There can be also a reverse process, where the algorithm decides that there is no justification for some data sources to be included in data processing as this increase the cost of operation. As a result the package guides the suggestion of actions to data acquisition module where sensor can be added or removed or to the user who can manually register new sensor node for more accurate monitoring.

4.4.4.2.3   Data analysing package

Data analysing package focus on real-time classification and decision algorithms applied to the collection of data supplied by sensors. Algorithms deployed on personal server determine the user's state and his or her health or well-being status and provide feedback to the user or a relevant third party either upon request or in any anomaly or life threatening event.

Taking advantage of recent improvements in processing power and battery life of modern PDA or smart phones the data analysing package will be based on artificial neural network engine. For this purpose the JOONE 2.0 (Java Object Oriented Neural Engine) (Joone 2007) will be ported to the Java ME platform. This solution that previously was reserved for PC machines with high computation power will be optimized to fit small portable systems. Porting JOONE will require the selection of most appropriate models that best answer the requirement of the application domain. The porting will need to anticipate some of the missing Java SE libraries in CLDC 1.1 and MIDP 2.0 package. Therefore substitution of these classes by the classes with similar functionality from these packages will be needed. Otherwise, the classes have to be implemented as part of `org.j2megloss` package. It is the developer package which "polishes" the J2ME libraries providing an implementation of some well know paradigms and classes from the Java SE platform.

4.4.4.2.4   Storage package

Data storage package represents the file system abstraction. It provides the personal server storage on flash memory card. It implements interfaces for writing and reading files as well as handles meta data operations such as create, rename or delete file(s).

4.4.4.2.5   Data distribution package

Data distribution package provides a decision tool for data analysing package in form of a mechanism capable to post alarms and notification remotely to third parties. It utilizes available WAN services and ensures successful information passing. It is a thin layer that brings data distribution logic in to the system that is responsible for accuracy and the quality of this service.

4.4.4.2.6   Presentation package

Presentation package is responsible for graphical user interface and real time data presentation to the user using various visualisation techniques such as charts plotting or simulations.

The GUI layer of the system is based on LWUIT (Lightweight User Interface Toolkit) (Knudsen 2008) which is a versatile and compact API for creating attractive application user interfaces for mobile devices. LWUIT provides sophisticated Swing-like capabilities without the tremendous power and complexity of Swing. Designed from the ground up as an efficient mobile user interface toolkit, LWUIT provides many useful Swing-like features. LWUIT offers a basic set of components, flexible layouts, style and theming, animated screen transitions, and a simple and useful event-handling mechanism. The toolkit is device independent and fully implemented using CLDC and MIDP packages. It has been tested and debugged on a variety of mobile devices.

Presentation package provides some customised or even only "polished" LWUIT controls that simplify the user experience. It provides also the port of JOpenChart Library and Toolkit (Müller 2002) to the CLDC 1.1. platform. It is a powerful and free Java Toolkit and library for embedding charts into different kinds of applications. Currently, the library provides all the functionality to draw different kinds of charts, like line, bar, pie and plot charts. Additionally, the class structure includes the necessary classes for the encapsulation of sets of data and the classes for all parts of a typical chart, like a coordinate system, coordinate axes, legends, title and chart renderers. It put great effort into a good design, i.e. understandable, logical class structures, elegant interfaces, use of abstract classes etc. The library is easily extendable to provide own chart functionalities. Data presentation package for the prototype system is only partially implemented including: custom widgets, MIDP 2.0 command prompt console and JOpenChart port.

Prototype data acquisition platform

5.1 Introduction

In the spirit of the system architecture presented in previous chapter, this chapter presents the prototypes of vital signs sensor devices designed and developed as part of this project. The aim of this chapter is to demonstrate how ubiquitous vital signs monitoring could be implemented. The prototype system includes an elastic chest strap sensor node, which integrates ECG, respiratory rate and temperature monitors, as well as an armband sensor node for pulse oximeter. Wearable sensor node devices capable of measuring and gathering long-term physiological recording were deployed using the SHIMMER platform (Burns, Greene, Mcgrath *et al.* 2010), which is a wireless sensor platform for ubiquities processing as well as wireless transmission of collected data over the Bluetooth network. Each sensor node includes a custom application as well as sensor specific extension boards and data processing algorithms which were evaluated. The personal server prototype, in turn, runs on a Bluetooth and WLAN/WWAN-enabled HTC HD2 (Htc Corporation 2010) smartphone device with Windows Mobile 6.5 operating system. It was implemented in line with guidance presented in system design chapter and serves as a network coordinator, data aggregator and integrated data processor.

5.2 SHIMMER prototyping platform

SHIMMER is a highly configurable and flexible wearable sensor platform, which can easily integrate and interact with existing technology and applications. It's flexibility rely on a free and open source, component-based operating system, called TinyOS (Philip Levis 2006). The TinyOS targets wireless sensor network (WSN) applications, allowing to program off-the-shelf sensors to meet exact data capture and transfer requirements. Shimmer platform enables to simplify the prototyping stage of wearable wireless sensor devices and immediately test them in a diverse range of applications from physiological signals monitoring, through user motion analysis, to biofeedback or rehabilitation, among other applications. It is highly configurable

thanks to a suite of technologies that offer compatibility with a wide range of sensors, wireless communication technologies such as: Bluetooth, 802.15.4 radio, or other useful features such as local storage to a microSD card, AnEx board for connection of self-designed or third party sensors or PIN outs on the internal and external expansion connectors, which are made available. On the other hand, the SHIMMER platform provides raw data streams, what allows a developer to have full control over the interpretation and analysis of sensed events.

Although the SHIMMER platform offers a wide range of extension boards and add-ons such as the ECG extension board or gyroscopes, it does not provide a body temperature, respiratory effort or pulse oximetry expansion modules. Therefore one of the challenges was to integrate above sensor with the SHIMMER platform and show how this could be achieved. For this purposes the available ADC channels on the Analog Expansion board (AnEx), have been used.

Figure 5.1 Shimmer platform in standard enclosure.

5.2.1   SHIMMER baseboard

The core element of the platform is the low-power 8 MHz MSP430F1611 (10Kbyte RAM, 48Kbyte flash) microprocessor (Texas Instruments 2006) from Texas Instruments™, which controls the operation of the device. The CPU configures and controls various integrated peripherals through I/O pins, some of which are available on the internal/external-expansion connectors.

Figure 5.2 SHIMMER baseboard interconnections and integrated devices (Shimmer Research 2010)

The most important component for data acquisition applications is an integrated 8-channel 12-bit analog-to-digital converter (ADC) which is used to capture sensor data from the built in 3-axis Freescale MMA7361 1.5/6g MEMs accelerometer (Freescale Semiconductor Inc. 2008) with selectable range, integrated tilt / vibration sensor, battery, or sensor expansions such as ECG, kinematics, GSR, and EMG. The external expansion allows communication on the serial port to and from the baseboard using the docking station. It features Hirose ST60 series 18 position rugged mobile style external header for charging, programming, and sensor extensions such as further discussed AnEx rapid prototyping expansion board (12 multi-purpose I/O connections) and Hirose DF12 series 20 position internal expansion header for internal sensor daughter boards (14 multi-purpose I/O connections).

For wireless data streaming the platform is equipped with class 2 Roving Networks RN-42 (Roving Networks 2010) Bluetooth module and 802.15.4 (ZigBee) Texas Instruments™ CC2420 radio modules. The Shimmer board has a built in microSD Flash slot for up to 2 GB additional storage. To improve usability, SHIMMER incorporates components to provide direct and immediate access to microSD flash memory using an external SD-flash card controller (SDHOST) for high-speed data transfer.

SHIMMER baseboard is powered with an integrated 280mAh Li-Ion battery. It features an integrated battery charger as well as scaled battery voltage measurement. Intelligent firmware through integrated battery management ensures low power communications and meets rigorous low power requirements. A push-button power controller powers off the entire board after a held press of the reset button. Software controlled power switching is provided for both the Bluetooth radio module and microSD socket. Three light-emitting diodes (LED) are used to display application status. All these features are compacted in a lightweight (22 grams) and very small, wearable form factor (53mm x 32mm x 15mm).

Figure 5.3 SHIMMER components layout (Shimmer Research 2010).

5.2.2 AnEx analog expansion board

Integration of respiratory effort sensor, temperature sensor and pulse oximeter with SHIMMER platform can be done with use of the Analog Expansion board (AnEx). It allows easy prototyping of third party sensors or custom sensing solutions.

The AnEx module enables an analog sensor, a digital output sensor, a serial UART, or a parallel bus interface to connect to SHIMMER. The AnEx board is connected to Shimmer via the external connector pin, and is positioned outside the Shimmer enclosure as presented in Figure 5.4. The AnEx board has a software assignable user control button, and can facilitate connection to an auxiliary power supply. It enables an access to +3VDC or +/-5VDC charge pump regulator with software-controlled enable pin.

Figure 5.4 SHIMMER Analog Expansion board (AnEx) (Shimmer Research 2010).

The AnEx board provides easy access to SHIMMER microcontroller gates using wired hole or solder pads. Figure 5.5 depicts the AnEx board layout and all 14 available pin connections. Because of the use of the analog sensors in development of proposed prototype

system, only four pins: power source and ADC input pins, were used. Used pins, their labels and description, are listed below:

- **+5**   +5V power source
- **G**   Ground
- **A0**   ADC input on channel 0
- **A7**   ADC input on channel 7

This image has been removed

Figure 5.5 AnEx board outline (Shimmer Research 2010).

5.3 Chest strap

The chest strap sensor node design, presented below, focuses on design of a respiratory rate monitor and a temperature monitor, as well as discusses the electrocardiogram signal conditioning (sampled using SHIMMER supplied ECG monitor). The prototype chest strap is made of an elastic material that twist around the chest. The material transfers chest movements to the piezoelectric sensor, that generates the sinusoidal signal, used for the assessment of respiratory rate. The temperature sensor, in turn, is sew in the elastic chest strap and placed in direct contact with skin to get accurate auxiliary temperature measurements. The SHIMMER baseboard unit is placed at breastbone and connects all sensors through touch proof pin interfaces.

5.3.1 ECG monitor

The ECG signal is being recorded with the use of the SHIMMER 3-lead ECG daughter board, which detects electrical impulses, traversing through heart muscles. The ECG daughter board connects to the internal connector on the SHIMMER baseboard and is enclosed within the SHIMMER enclosure. The ECG daughter card provides four touch proof pins, which allow to connect ECG leads. Leads, in turn, connect the monitor with four conventional disposable electrodes attached to the chest of the patient. The unique AC-coupled topology with clamping diodes provides excellent transient recovery. Moreover, through the use of low offset precision amplification stage, with ~15pA of leakage for high impedance matching, the ECG daughter card can also be used during the exercise, in order to provide information on heart's response to physical activity.

Figure 5.6 SHIMMER ECG daughter board (Shimmer Research 2010).

The SHIMMER ECG daughter board has been validated by Mcgrath and Dishongh (2009). It was demonstrated that the SHIMMER ECG daughter boards can deliver ambulatory quality ECG signal for research application purposes. Moreover, providing quick recovery from movement artefacts the SHIMMER ECG offers a cost effective wireless ECG monitoring system that captures information to highest industry standards. It has also received CE certification ascertaining it meets EU consumer health and safety requirements for medical equipment.

The sampling frequency of the ECG monitor can be set programmatically, independently from any other extension boards, that are deployed on the same sensor node. Normally the frequency of the ECG signal ranges from 0.5-100Hz while its dynamic amplitude range is ±5 mV (Yong and Jiang Hong 2004, Saritha, Sukanya and Murthy 2008). From the Nyquist sampling theorem, whihc provides a prescription for the nominal sampling interval required to avoid aliasing, the sampling frequency should be at least twice the highest frequency contained in the signal (Vaidyanathan 2001). In mathematical terms:

$$f_s \geq 2f_c \qquad (9)$$

where f_s is the sampling frequency (how often samples are taken per unit of time or space), and f_c is the highest frequency contained in the signal. Aliasing arises when a signal is discretely sampled at a rate that is insufficient to capture the changes in the signal. If the aliasing occur the original signal's information may not be completely recoverable from the sampled signal, resulting in the signal reconstructed from samples being different from the original continuous signal. Therefore, according to the Nyquist sampling theorem and the ECG frequency spectrum, the sampling frequency for the ECG monitor is set to 200 Hz, what is double the highest frequency of the signal of interest. Such sampled signal is of good quality for further analysis such as computation of heart rate, heart rate variability analysis or classification of heart beats.

5.3.1.1 ECG calibration

The amplitude of the ECG signal is a voltage difference measured in millivolts (mV), which is recorded between two metal plates or electrodes on the surface of the body, usually on two limbs. Since ECG electrodes detect very small analog signal, so a key element in the signal

chain is the analog front end (AFE), consisting of several amplifiers and analog-to-digital (ADC) converters. The first one increase the signal amplitude by the factor K to match the full scale measurement range of the analog-to-digital converter (ADC). The ADC converts a continuous physical signal, in this case voltage, to a digital number that represents the amplitude's quantity. It quantises and records the signal with predefined digital resolution N over the measurement voltage range of V_{low} to V_{high}. The digital resolution of the converter specifies the number of discrete values, usually expressed in bits, that it can produce over the range of analog values.

The ECG monitor calibration process requires to adjust the amplitude scale and units of the recorded digital signal to its original physical signal. This can be achieved by correcting the digital signal by offset, gain and voltage resolution. Firstly, we have to calculate the ADC voltage resolution (sensitivity). This is given by the following equation (10).

$$ADC_{res} = \frac{V_{high} - V_{low}}{2^N - 1} \tag{10}$$

For example, the voltage range for the SHIMMER ADC set to 3V and digital resolution of 12 bits impose the ADC voltage resolution to be (3V-0V)/4095 ≈ 7.326x10^{-4} V ≈ 0,73 mV, what is the interval between consecutive digital discreet values of the ADC output.

Secondly, we have to measure the operational amplifier (op-amp) gain K of the ECG monitor. This is defined as ratio of the signal output of the ADC to the signal input of the same ADC, expressed in the same units of measurement, in this case millivolts (mV). Since an ideal op-amp is hypothetical, the physical op-amps strive to achieve same characteristics. Although physical op-amps come very close to the ideal op-amp, but no op-amp can actually achieves the perfection of an ideal op-amp. Therefore the actual gain correction is specific to each hardware realisation. In practice it is measured by applying the sin wave of known amplitude and frequency to electrodes and measuring the output on the ADC. By convention the offset-corrected maximum digital signal $ADC_{out}(i)$, measured for the giving maximum input signal $V_{in}(t)$ is used to calculate the gain using equation (11),

$$K = \frac{V_{out}}{V_{in}} = \frac{[\max(ADC_{out}(i)) - ADC_{offset}] \cdot ADC_{res}}{\max(V_{in}(t))} \tag{11}$$

where the ADC_{offset} is the digital ADC output, that corresponds to 0 mV on the system input. The default value for the ECG daughter board gain is 175 and ADC offset is 2060. For most applications these values are sufficiently accurate, but exact value is the subject to environmental and component variation.

Knowing the ADC voltage resolution, gain and offset the original ECG signal can be rebuild according to the following equation (12).

$$ECG(t) = \frac{(ADC_{out}(i) - ADC_{offset}) \cdot ADC_{res}}{K} \tag{12}$$

5.3.1.2   Noise cancelation of ECG signal

Accurate ECG monitoring is difficult to achieve when the monitoring is performed during normal daily activity. It is mainly due to high noise contamination caused by (Yong and Jiang Hong 2004):

- power line interference, caused by environmental interference with the ECG device
- baseline drifts changes or baseline wander, as a result of respiration and chest movements,
- motion artefacts, as a result of subject movements,
- muscular/skeletal noise, as a result of skeletal muscle contractions, that generate bioelectrical potentials – also known as Electromyography (EMG) signal – used to evaluate the electrical activity produced by skeletal muscles,
- or morphological differences in the waveform between subjects.

Examples of such noise contaminated ECG signals are presented in Figure 5.7.

a)   b)   c)

Figure 5.7 Examples of noise contaminated ECG: a) baseline drifts changes, b) motion artefacts, c) muscular/skeletal noise

By using off-the-shelf SHIMMER ECG daughter board rather than designing one from scratch, the noise reduction tasks can only be achieved with help of digital filters. This, however increases the complexity of ECG detection, what is crucial when such analyses should be performed in a timely manner. Therefore, the choice of filters used by the prototype was made base on their suitability for the application but trying to balance between noise reduction requirements and computational load.

Cancelling noise present in ECG signals can be achieved through different digital filtering techniques depending on the noise source. Both linear time-invariant filters as well as adaptive linear time-variant filters are applicable for this purpose, but their selection is governed by a set of principles.

The linear time-invariant digital filtering is suitable when the noise is a stationary random process that is statically independent of the signal. Power line interference, baseline drifts changes and muscular/skeletal noise have these characteristics. To remove this interference, a low-pas, high-pass, or band-reject filters are used at the various cut-off frequencies.

Adaptive linear time-variant filtering, on the other hand, is a good option when removing motion artefacts, because motion artefacts are neither stationary nor necessarily random. The adaptive filter minimises the error between primary input of the noisy ECG and reference input r(n), such, which is associated with motion artefacts. Because motion-free ECG signal has little correlation with subject motion, the reference input r(n) such as accelerometer signal, which has

little correlation with primary input x(n), provides good information about the noise, that should be removed from the input signal.

To assess the noise reduction performance of these methods for ECG monitoring during normal daily activities, representative records from MIT-BIH Noise Stress Test Database (nstdb) (Moody, Muldrow and Mark 1984) as well as data collected by the SHIMMER sensor system were tested under noise conditions outlined in Figure 5.7. The nstdb ECG recordings were created by the script nstdbgen (Moody 2014) using noise free ECG reference signal 118 from the MIT-BIH Arrhythmia Database (mitdb) (Moody and Mark 2001), to which calibrated amounts of baseline wander (in record 'bw') and muscle (EMG) artifact (in record 'ma') noise was added beginning after the first 5 minutes of each record, during two-minute segments alternating with two-minute clean segments. The signal-to-noise ratios (SNRs), defined in (Moody 2014), during the noisy segments of record 118, used for this evaluation, are:

- 6db SNR for baseline wander reduction (bw),
- 18db SNR for muscle (EMG) artefact (ma).

For these records both qualitative visual inspection of excerpt from ECG records and the quantitative measure of Signal-to-Noise Ratio (SNR) was applied and presented below. This definition of SNR used to evaluate signal before and after the noise reduction, differs however from the one used in nstdbgen script. Because the signal and the noise were measured across the same impedance, then the SNR (expressed using the logarithmic decibel scale) was obtained by calculating the square of the amplitude ratio as in equation (13) below.

$$SNR = 10 \log_{10} \frac{P_{signal}}{P_{noise}} = 10 \log_{10} \left(\frac{A_{signal}}{A_{noise}} \right)^2 = 20 \log_{10} \left(\frac{A_{signal}}{A_{noise}} \right) \tag{13}$$

Where P is average signal power and A is the Root Mean Square (RMS) amplitude of the signal defined by equation (14) as:

$$A_x = RMS(x) = \sqrt{\frac{1}{n} \left(x_1^2 + x_2^2 + \ldots + x_n^2 \right)} \tag{14}$$

Using equation (13) the SNRs 'after the noise reduction' and 'before the noise reduction' were compared by calculating the SNR changes (donated as SNR increase), using the following equation (15),

$$SNR_{increase} = SNR_{after} - SNR_{before} =$$
$$= 20 \log_{10} \left(\frac{RMS(ECG_{ref})}{RMS(ECG_{ref} - ECG_o)} \right) - 20 \log_{10} \left(\frac{RMS(ECG_{ref})}{RMS(ECG_{ref} - ECG_i)} \right) \tag{15}$$

where ECG_i and ECG_o are respectively, the ECG signal before and after the motion artifacts reduction. The equation (15) can further be simplified using logarithm identity (16),

$$\log_a \left(\frac{x}{y} \right) = \log_a(x) - \log_a(y) \tag{16}$$

to be (17) as follows.

$$SNR_{increase} = SNR_{after} - SNR_{before} = 20 \log_{10} \left(\frac{RMS(ECG_{ref} - ECG_i)}{RMS(ECG_{ref} - ECG_o)} \right) \qquad (17)$$

5.3.1.3   Linear time-invariant digital filter design

Digital filters are software realisations of linear or non-linear mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. There are two categories of digital filters: the recursive filters, often referred to as infinite impulse response (IIR), and the non-recursive filters, known as finite impulse response (FIR) filters. Figure 5.8 illustrates the block diagrams for both filter types. The main difference between IIR and FIR is that the second type do not normally have the feedback (a recursive part of a filter), even though recursive algorithms can be used for FIR filter realisation. Example of such recursive FIR implementation is the moving average filter, in which the n-th prior sample is subtracted (fed back) each time a new sample comes in.
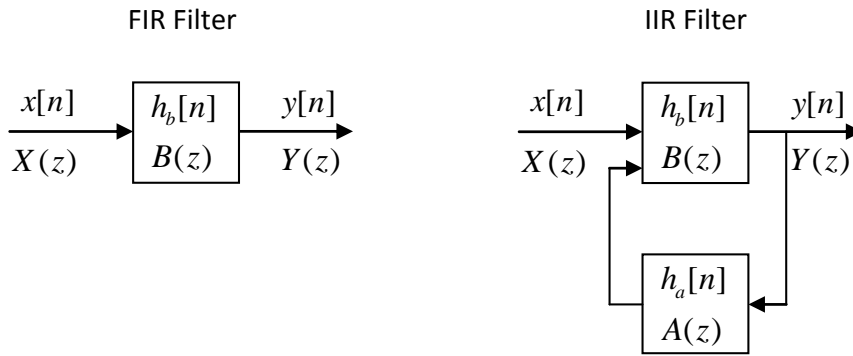
Figure 5.8 Block diagrams of FIR and IIR filter.

In practice however, FIR filters are more often used then IIR due to their unique properties. First property is that they do not require a feedback, what makes them simple to implement. In practice, all DSP filters must be implemented using finite-precision arithmetic, that means, with a limited number of bits. The use of finite-precision arithmetic in IIR filters can cause significant problems due to the use of feedback, while FIR filters without feedback can usually be implemented using fewer bits by looping a single instruction. Second property, is the FIR filter linear-phase response, which delay the input signal but don't distort its phase. Third property is that FIR filters are inherently stable, therefore time-invariant. Since they have no feedback elements, any bounded input results in a bounded output (Meddins 2000).

For a discrete-time FIR filter, the output is a weighted sum of the current and a finite number of previous values of the input. The operation is illustrated in Figure 5.9 and defined by the equation (18), which determine the output sequence $y[n]$ by convolving its input signal $x[n]$ with its impulse response b.

$$y[n] = b_0 x[n] + b_1 x[n-1] + \ldots + b_N x[n-N] = \sum_{i=0}^{N} b_i x[n-i] \qquad (18)$$

where: $x[n]$ is the input signal at sample n,

$y[n]$ is the output signal at sample n,

b_i are the filter coefficients, which for FIR filter equals to impulse response $b_i = h[i]$,

N is the filter order
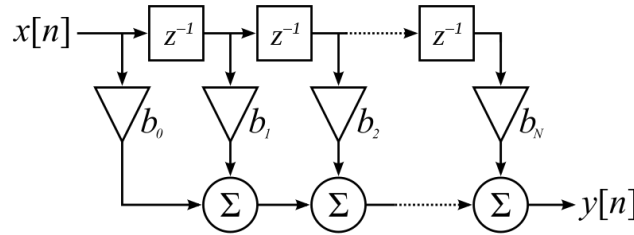
Figure 5.9 FIR filter direct realisation

A digital filter is characterised by its transfer function, or equivalently, its difference equation. FIR filters can be designed using different methods, but most of them are based on ideal filter approximation by the window function as illustrated in Figure 5.10. The objective is not to achieve ideal characteristics, as it is impossible anyway, but to achieve sufficiently good characteristics of a filter. As such, designing a filter consists of developing specifications appropriate to the problem, and then producing a transfer function, which meets the specifications. The characteristics of the transfer function as well as its deviation from the ideal frequency response depend on the filter order and window function in use. The transfer function of FIR filter approaches the ideal as the filter order increases, thus increasing the complexity and amount of time needed for processing input samples of a signal being filtered. As such the balance needs to be achieved between the accuracy and the computational load.

Figure 5.10 Ideal low-pass filter approximation (Milivojević 2009)

The transfer function can be found via the Z-transform of a FIR filter frequency response. The Z-transform is performed upon discrete-time signals and converts a discrete

time-domain signal into a complex frequency-domain representation. The transfer function of discrete-time system, for both FIR and IIR types is defined as:

$$H(z) = \frac{\sum_{i=0}^{M-1} b_i z^{-1}}{\sum_{j=0}^{N-1} a_j z^{-j}} \tag{19}$$

where: b_i and a_j are the feed forward (non-recursive part) and feedback (recursive part) filter coefficients, respectively. As previously discussed, the FIR filter does not have this recursive (feedback) part of the transfer function, so the expression above can be simplified as:
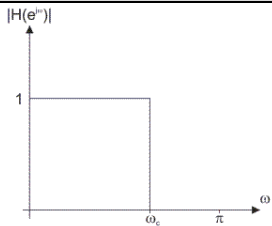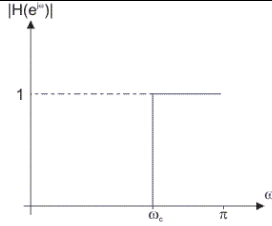
$$H(z) = \sum_{n=0}^{N-1} h[n] z^{-n} \tag{20}$$

where $h(n)$ is the impulse response of discrete-time system. It is defined as the output signal that results when an impulse, a short-duration time-domain signal (typically Kronecker delta function $\delta[n]$ for discrete systems) is applied to the system input. It characterise the system's output in the time domain as opposed to frequency response.

The crucial step in the FIR filter design process is to compute the impulse response coefficients of the intended filter. The simplest and most common approach is the window method. It takes the ideal filter coefficients (ideal filter impulse response), denoted as $h_d[n]$ and truncate it to match with filter order. This means multiplying it by the window function coefficients $w[n]$ that satisfy the given specifications as in equation (21).

$$h[n] = w[n] \cdot h_d[n] \tag{21}$$

The four standard ideal filter coefficients (ideal filter impulse responses) and their transfer functions are contained in the Table 5.1, where values of the variable n, range between 0 and N, where N is the filter order with constant M expressed as $M = N/2$.

Filter Type	Transfer Function	Ideal filter coefficients $h_d[n]$
Low-pass filter		$h_d[n] = \begin{cases} \dfrac{\sin(\omega_c(n-M))}{\pi(n-M)} ; n \neq M \\ \dfrac{\omega_c}{\pi} ; n = M \end{cases}$
High-pass filter		$h_d[n] = \begin{cases} 1 - \dfrac{\omega_c}{\pi} ; n \neq M \\ -\dfrac{\sin(\omega_c(n-M))}{\pi(n-M)} ; n = M \end{cases}$
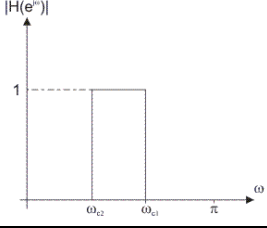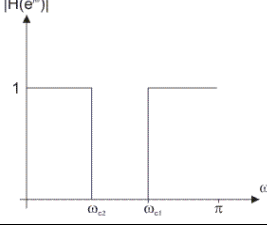
| Band-pass filter | | $$h_d[n] = \begin{cases} \dfrac{\sin(\omega_{c2}(n-M))}{\pi(n-M)} - \dfrac{\sin(\omega_{c1}(n-M))}{\pi(n-M)} \; ; n \neq M \\[2ex] \dfrac{\omega_{c2} - \omega_{c1}}{\pi} \; ; n = M \end{cases}$$ |
| Band-stop filter | | $$h_d[n] = \begin{cases} \dfrac{\sin(\omega_{c1}(n-M))}{\pi(n-M)} - \dfrac{\sin(\omega_{c2}(n-M))}{\pi(n-M)} \; ; n \neq M \\[2ex] 1 - \dfrac{\omega_{c2} - \omega_{c1}}{\pi} \; ; n = M \end{cases}$$ |

Table 5.1 The impulse responses of four standard ideal filters (Orfanidis 1995).

A window function is a finite array consisting of coefficients that are zero-valued outside of some chosen interval selected to satisfy the desirable requirements. When designing digital FIR filters using window functions the point is to find these coefficients denoted as $w[n]$. To do so, special attention should be paid to window function selection as each function is a kind of compromise between the selectivity expressed as the length of the transition region (the frequency range between the pass band and the stop band of a digital filter) and the level of suppression of undesirable spectrum, expressed as stop band attenuation (an amplitude loss, usually measured in dB). Amongst wide spectrum of functions available to choose from for the ECG analysis the Hann window function best met the requirements and achieved the best results in undesired noise cancelation.

The Hann window function is an optimal compromise between requirements for as narrow transition region as possible and as higher stop band attenuation as possible. The Hann window coefficients are expressed in the following equation (22).

$$w[n] = \frac{1}{2}\left[1 - \cos\left(\frac{2\pi n}{N-1}\right)\right] \tag{22}$$

where $0 \leq n \leq N-1$. The following figure 5.11 illustrates the Hann window coefficients in both time and frequency domain.

Figure 5.11 The Hann window coefficients in: a) the time domain; b) the frequency domain (Milivojević 2009)

By looking at figure 5.11, it can be observed that the first lobe in the frequency domain of this filter has attenuation of 31dB, whereas for the whole frequency spectrum it adds to attenuation of 44dB. The filter is also characterise by relatively fast increase in the stop band attenuation of the following lobes. Moreover the transition region for the Hann filter is the same as for triangular window, which has the sharpest fall, however with poor attenuation. Therefore for minimum attenuation, the Hann window will have a narrower transition region, what makes this filter one of the most desirable for application to ECG signal filtering.

Summarising, the FIR filter design process can be split into several steps:

1. Define filter specifications;
2. Specify a window function according to the filter specifications;
3. Compute the filter order required for a given set of specifications;
4. Compute the window function coefficients;
5. Compute the ideal filter coefficients according to the filter order;
6. Compute FIR filter coefficients according to the obtained window function and ideal filter coefficients;

While designing the filter for certain applications, the visual inspection of the filter output is required, in order to establish the most optimal transition regions. This can be done by increasing or decreasing the filter order and repeating steps 4, 5 and 6 for each new order value as many times as needed in order to obtain satisfactory results for each noise type. The following sections discuss such filter selection processes for each ECG noise type.

5.3.1.3.1 Power line inference reduction

Power line interference is typically caused by environmental interference with the ECG device that comes from electromagnetic fields from power lines that can cause 50/60 Hz sinusoidal interference depending on the Alternating current (AC) source. It can also be accompanied by some of its harmonics. Such noise can cause problems interpreting low-amplitude waveforms and spurious waveforms can be introduced. Precautions should be taken to keep power lines as far as possible or shield and ground them, but this is not always possible. Therefore signal post-processing is required to remove this noise. For this purposes the ECG monitoring systems use a FIR band-pass filter with Hann window and cut-off frequencies set to 55 and 65 Hz. However as further discussed, due to the filters used in this project to eliminate baseline wander and muscular/skeletal noise, the power line inference dedicated filter is not required. This is because power line inference frequency band falls out of the narrow pass band of those filters.

5.3.1.3.2 Baseline wander reduction

Baseline wander, or extraneous low-frequency high-bandwidth components, can be caused by such sources as: perspiration (effect of electrode impedance), respiration and some body movements. Second excerpt in Figure 5.12 shows the excerpt of the ECG signal from record 118 of the MIT-BIH Noise Stress Test Database containing baseline wander noise. Such

noise can cause problems in the analysis, especially when examining the low-frequency ST-T segments. Baseline wander frequency is typically less than 0.5 Hz, except for the abrupt shifts due to motion artefacts (discussed in Section 5.3.1.4.1).

Proposed solution to this problem is to use the FIR high pass filter to cut-off the lower-frequency components of the signal at about 0.5 Hz. Such cut-off frequency is defined by the slowest acceptable heart rate. The heart rate can drop to 40 bpm, implying the lowest frequency of the clean ECG signal to be 0.67 Hz. By setting the cut-off frequency to 0.5 Hz it ensures that the interference of the FIR filter with the desired signal frequency band is avoided. The filter is realised with the Hann window due to its very low aliasing characteristic.

To test the noise suppression capabilities of this cut-off frequency, the 100-th order high-pass filter with Hann window was implemented and tested on record 118 from mitdb database contaminated by the baseline wander noise (bw) from nstdb database. The signal-to-noise (SNR) ratio for the test records was set to 6 dB (according to signal-to-noise ratio definition used in (Moody 2014)) what corresponds to -15.2dB SNR using the SNR definition used in this project. Figure 5.12 presents the excerpts from the ECG signal before and after filtering, accompanied by noise and ECG reference signals.

ECG signal & Baseline Wander (BW) noise
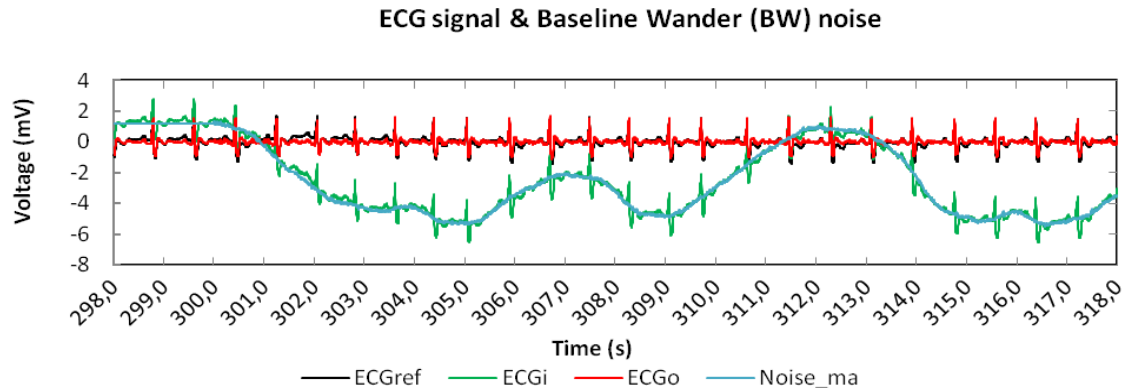
Figure 5.12 Twenty seconds ECG signal and BW noise excerpts from mitdb/118 and nstdb/bw records respectively

RMS	Noisy (bw) segments of the ECG signal (mitdb/118) [mm:ss]						Total	Total*
	5:00-7:00	9:00-11:00	13:00-15:00	17:00-19:00	21:00-23:00	25:00-27:00		
ECG_{ref}	0.411	0.401	0.397	0.408	0.418	0.400	0.406	0.409
ECG_i	3.167	5.320	10.150	1.691	2.134	2.421	5.076	2.414
ECG_o	0.324	2.192	5.144	0.322	0.324	0.340	2.298	0.327
$ECG_{ref} - ECG_o$	3.107	5.325	10.197	1.624	2.082	2.376	5.076	2.360
$ECG_{ref} - ECG_i$	0.226	2.196	5.199	0.215	0.222	0.222	2.311	0.221
SNR [dB]								
SNR_{before}	-17.6	-22.5	-28.2	-12.0	-14.0	-15.5	-21.9	-15.2
SNR_{after}	5.2	-14.8	-22.3	5.6	5.5	5.1	-15.1	5.3
$SNR_{increase}$	22.8	7.7	5.9	17.6	19.4	20.6	6.8	20.6

*) excl. 9:00-11:00 and 13:00-15:00 segments of reference signal contaminated by high frequency subject inherent noise

Table 5.2 Signal-to-Noise Ratios for the Baseline wander noise contaminated ECG reference record mitdb/118 before and after the application of the 0.5 high-pass filter.

Table 5.2 summarises the results obtained during the tests on the first-channel of ECG reference record 118 from the mitdb database, contaminated by the baseline wander noise (bw) from nstdb database. As it can be seen the total SNR for noisy segments (excluding segments 9:00-11:00 and 13:00-15:00 of reference signal which are contaminated by subject inherent noise) has significantly improved by 20.6 dB from - 15.2dB before to 5.3dB after filtering. Figure 5.13 presents ten seconds excerpts from the 118 record. As it can be seen, the proposed solution successfully removed the whole baseline wander noise with no effect on the original signal amplitude.
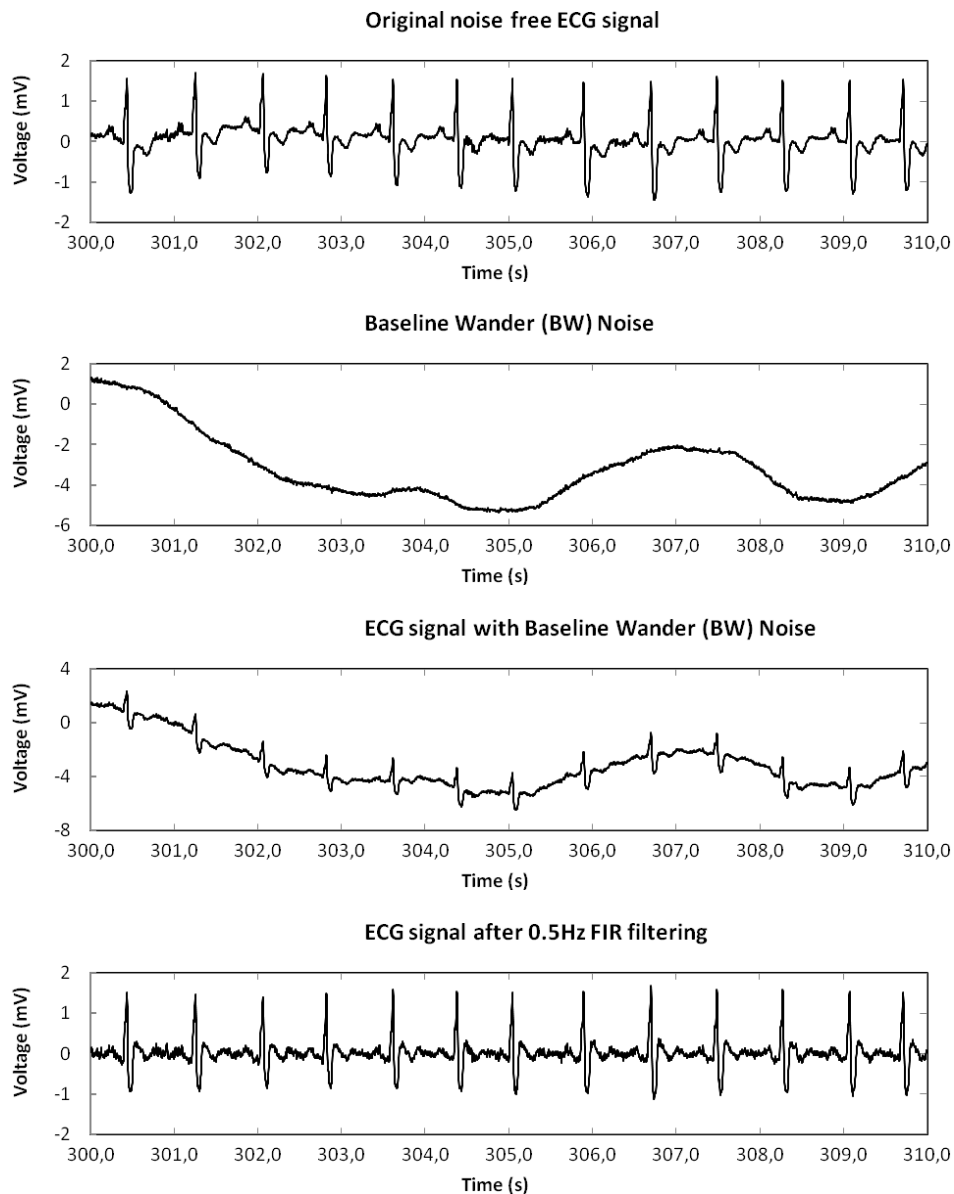
Figure 5.13 Ten seconds excerpts from mitdb database ECG Record 118 before and after the 0.5 Hz FIR filtering.

5.3.1.3.3 Muscular/skeletal noise reduction

The heart is not the only thing in the body that produces measurable electricity. When the body moves the skeletal muscles undergo tremors which "bombard" the ECG signal with

126

seemingly random, high frequency activity causing muscular/skeletal noise. This electrical interference is caused by bioelectric potentials, which constitute the Electromyogram abbreviated as EMG. These potentials may be measured at the surface of the body near a muscle of interest. Such noise can cause severe problems to ECG monitoring especially in recordings during exercise as low-amplitude ECG waveforms can be obstructed.

EMG measurements are intended to obtain an indication of the amount of activity of a given muscle, or group of muscles, rather than that of an individual muscle fiber. Thus, the pattern is usually a summation of the individual action potentials from the fibers, that build the muscle or muscles being measured. The action potential of a given muscle (or nerve fiber) has a fixed magnitude, regardless of the intensity of the stimulus that generates the response. Thus, the muscle contraction intensity does not increase the net height of the action potential pulse but does increase the rate at which each muscle fiber fires and the number of fibers that are activated at any given time. The amplitude of the measured EMG waveform is the instantaneous sum of all the action potentials generated by fibers at any given time. Thus, the EMG waveform appears very much like a random-noise waveform, that varies in amplitude with the amount of muscular activity. Peak amplitudes vary from 25 μV to about 5 mV, depending on the location of the EMG electrodes with respect to the muscle and the activity of the muscle. However, in practice muscle tremors are often a lot more subtle than 5mV. In terms of frequency response, the full frequency band for the EMG signal varies from about 5 Hz to well over 500 Hz (Van Boxtel 2001). In practice its most distorting frequencies for ECG signal start at about 20Hz (Dotsinsky and Mihov 2008). It should be noted at this stage that, as recommended by the American Heart Association (AHA) and American College of Cardiology, the diagnostic quality ECG frequency band varies between 0.05 and 150 Hz (Kligfield, Gettes, Bailey *et al.* 2007) however its practical frequency band for monitoring-quality ECGs, used in ubiquities monitoring, by convention is limited to 0.5-40Hz (Venkatachalam, Herbrandson and Asirvatham 2011). The fundamental frequency of the QRS complex at the body surface is ≈10 Hz and the fundamental frequency of T waves is approximately 1 to 2 Hz. Therefore most of the diagnostic information of the ECG signal for adults, is contained below 100 Hz. Moreover, ECG signal filtered to 1 to 30 Hz frequency band, produces a stable ECG that is generally free of artefacts. Such signal frequency bandwidth is acceptable for QRS analysis, however unacceptable for clinically significant diagnosis, because of distortions to both high- and low-frequency components of the signal (Kligfield, Gettes, Bailey *et al.* 2007). Therefore, since both EMG and ECG practical frequency responses overlap, thus, the filtering prior to QRS detection is required.

When considering real-time, non-clinical QRS analysis within the practical ECG frequency band, the filtering of the muscle tremors can be relatively easy to obtain with a low-pass filter, at a relatively low computational cost. According to the literature, suggested minimum cut-off frequency is 35 Hz. However, since the most of the practical ECG spectrum locates below 30 Hz and EMG relatively wide high frequency spectrum starts at 20 Hz, thus in

this work the minimum cut-off frequency of the low-pass filter is decreased to 25 Hz, what is the middle-frequency of the overlapped EMG and ECG frequency ranges.

To test the noise suppression capabilities of this cut-off frequency, the 100-th order low-pass filter with Hann window was implemented and tested on record 118 form mitdb database, contaminated by the muscle (EMG) noise (ma) from nstdb database. The signal-to-noise ratio for the test record was set to 18 dB, according to Moody's definition of signal-to-noise ratio, what corresponds to -5.7dB SNR, according to the definition proposed in this work. Such SNR allows to simulate a moderate noise contaminated ECG signal. Figure 5.14 shows the excerpt from these two records.
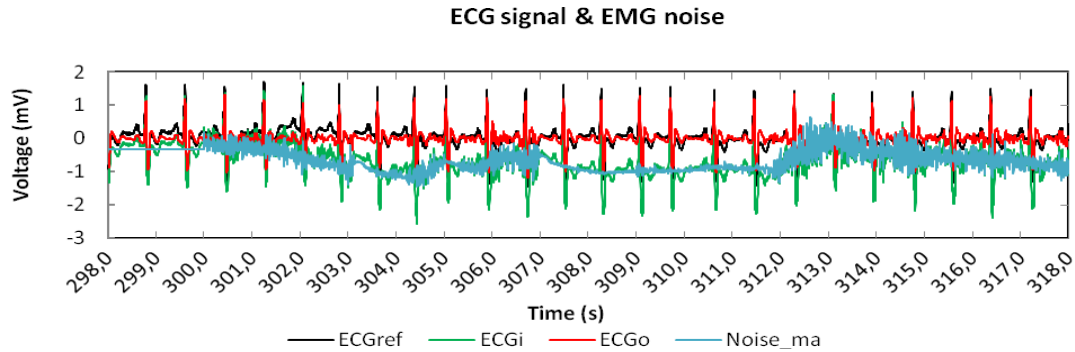
Figure 5.14 Twenty seconds ECG signal and EMG noise excerpts from mitdb/118 and nstdb/ma records respectively.

RMS	Noisy (ma) segments of the ECG signal (mitdb/118) [mm:ss]						Total
	5:00-7:00	9:00-11:00	13:00-15:00	17:00-19:00	21:00-23:00	25:00-27:00	
ECG_{ref}	0.411	0.401	0.397	0.408	0.418	0.400	0.406
ECG_i	0.908	0.802	1.174	0.743	0.834	0.816	0.891
ECG_o	0.289	0.276	0.293	0.289	0.293	0.289	0.288
$ECG_{ref} - ECG_i$	0.773	0.711	1.085	0.650	0.709	0.702	0.785
$ECG_{ref} - ECG_o$	0.250	0.256	0.213	0.239	0.247	0.223	0.238
SNR [dB]							
SNR_{before}	-5.5	-5.0	-8.7	-4.0	-4.6	-4.9	-5.7
SNR_{after}	4.3	3.9	5.4	4.6	4.6	5.1	4.6
$SNR_{increase}$	9.8	8.9	14.1	8.7	9.2	10.0	10.4

Table 5.3 Signal-to-Noise Ratios for the muscle (EMG) noise contaminated ECG reference record mitdb/118 before and after the application of the 25Hz low-pass filter.

Table 5.3 summarises the results obtained during the tests which shows that the total SNR for noisy segments has improved by 10.4 dB from -5.7 dB before to 4.6 dB after filtering. The ECG signal excerpts, presented in Figure 5.15, illustrates how the contaminated signal was obtained. A conditionally clean ECG episode (upper trace) is mixed with tremor (second trace) to be used further (third trace) as the filter input. Noise suppression of the contaminated record 118 episode is presented in the lower trace. By comparing the clean ECG episode (upper trace) with the filter output (lower trace) it can be noticed that much of the noise has been successfully removed, while the useful ECG frequency band has been preserved. The results shows that the proposed low-pass filter allows to eliminate large portion of the most common tremors, in ubiquities applications where high frequencies of ECG are not part of the analysis. However, in

this way the amplitudes of sharp QRS waves have been reduced at average by -2dB, what needs to be considered when design an ECG beat classification algorithms.
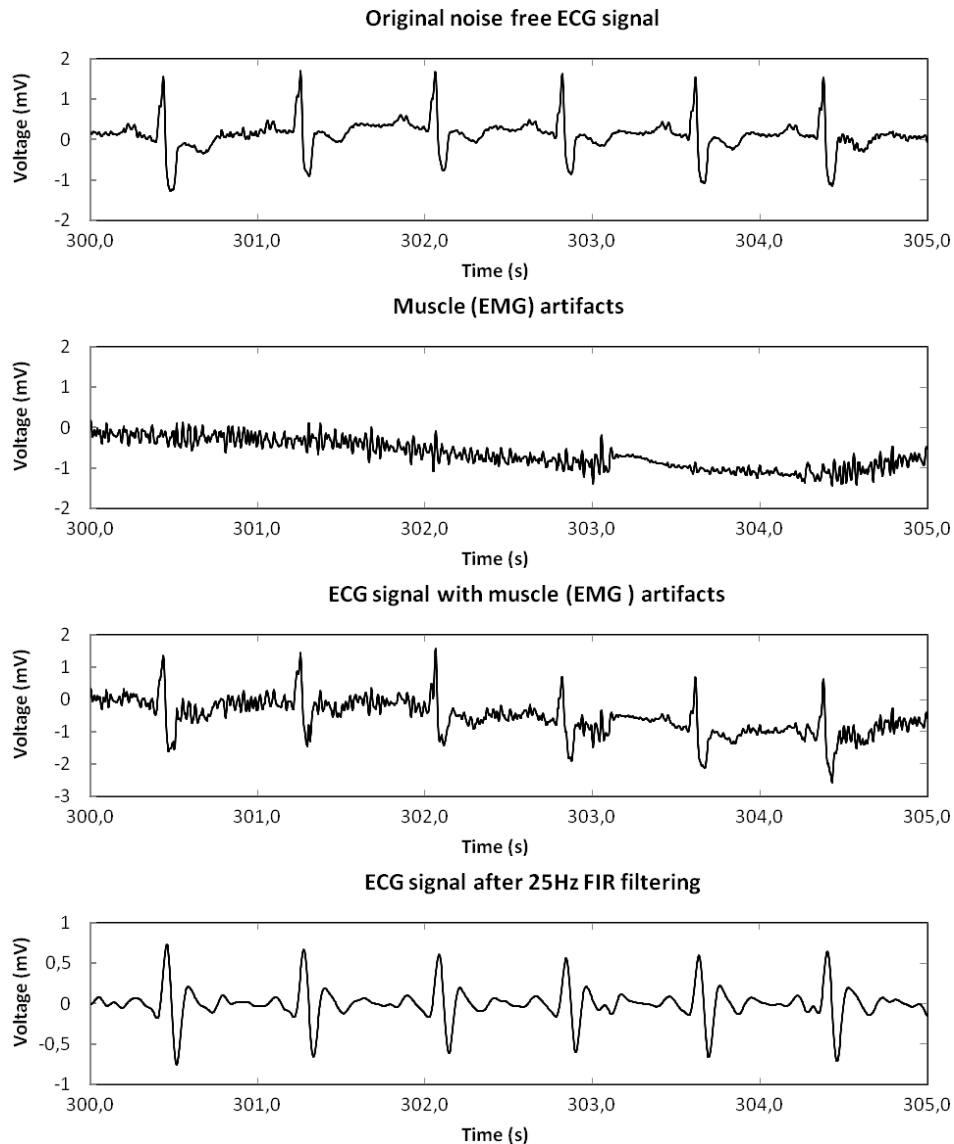
Figure 5.15 Five seconds excerpt from MIT-BIH Arrhythmia Database ECG Record 118 with muscle (EMG) artefacts from MIT-BIH Noise Stress Test Database before and after the 25 Hz FIR filtering.

5.3.1.4   Adaptive linear time-variant filter design

An adaptive filter is a digital filter that iteratively adjust itself to the unknown input signal. This ability to operate in an unknown environment and track time variations of input statistics makes adaptive filters a powerful tool in signal processing and control applications. In comparison, a non-adaptive filter has static filter coefficients, which require a priori information about the statistical characteristics of the input data and is not adequate to be applied for non-stationary inputs.

Adaptive noise cancelling is one of the applications of adaptive filters. It is a method of estimating signals corrupted by some unknown and additive noise or interference, with the

cancellation process being optimized over time. Practical real-time realisation of adaptive filter requires two inputs called a primary sensor and a reference sensor. The primary sensor supplies a desired signal of interest buried in noise The reference sensor, in turns, supplies a noise reference signal. There are two assumptions here: (1) the signal and noise at the output of the primary sensor are uncorrelated; and (2) the signal at the output of the reference sensor is correlated with the noise component of the primary-sensor output. In such settings the noise is adaptively filtered and subtracted from the primary input to obtain the signal estimate. In the adaptive noise cancelling system, the system output serves as the error signal for the adaptive process. With a proper algorithm, the filter can operate under changing conditions and readjust itself continuously to minimize the error signal.

An adaptive filter self-adjusts the filter coefficients (transfer function) according to an adaptive algorithm, driven by the error signal. The block diagram, shown in the figure 5.16, serves as a foundation for noise cancelling adaptive filter realisation,

Desired Signal + Noise

$d[k] = s[k] + n[k]$

$e[k] = d[k] - y[k] = \hat{s}[k]$

Error

$y[k] = \hat{n}[k]$

$x[k]$

ALC

Correlated with $n[k]$

After convergence $\quad e[k] = d[k] - y[k] = s[k] + n[k] - \hat{n}[k] = \hat{s}[k]$
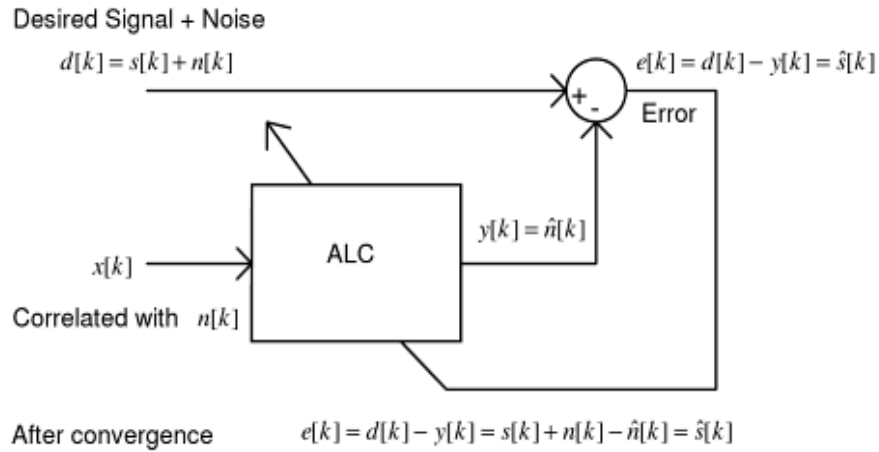
Figure 5.16 Typical adaptive filter

where: $x[k]$ is the noise reference input signal to a linear filter at sample k,

$y[k]$ is the corresponding output signal at sample k,

$d[k]$ is an primary, noise corrupted input signal to the adaptive filter at sample k,

$e[k]$ is the error signal (signal of interest) at sample k, that denotes the difference between $d[k]$ and $y[k]$.

Perhaps the most popular method for adjusting the coefficients of an adaptive filter with such cost function are the least mean squares (LMS) and recursive least squares (RLS) algorithms. In this work we use the Normalised LMS (NLMS) which is the modified version of the LMS algorithm. The NLMS algorithm updates the coefficients of an adaptive filter by using the following equation (23):

$$\mathbf{w}[k+1] = \mathbf{w}[k] + \mu \cdot e[k] \cdot \frac{\mathbf{x}[k]}{\left\|\mathbf{x}[k]\right\|^2} \tag{23}$$

where: μ is the step size of the adaptive filter,

$w[k]$ is the filter coefficients vector at sample k,

$x[k]$ is the filter input vector at sample k,

$\left\|x[k]\right\|$ is the Euclidian norm of input vector at sample k.

Concluding, the NLMS algorithm performs the following sequence of operations to update the coefficients of an adaptive filter:

1) Calculate the output signal $y[k]$ from the FIR filter.

2) Calculate the error signal $e[k]$.

3) Update the filter coefficients.

5.3.1.4.1 Motion artefacts reduction using 3-axis accelerometer

ECG signal monitoring during usual daily activities, involve such adverse situations as walk or exercises. Therefore the signal obtained is often contaminated by motion artefacts caused by the patient's movements. Any motion can produce a change in skin potential due to skin stretching or a change in the position of the conductive fabric electrodes, what in turn affects the quality of the ECG signal. Considering the variable frequency response and amplitude of this noise, the conventional ECG enhancement methods based on static finite impulse response (FIR) filters are impractical for this application. Their use, for instance with very narrow pass-band, would introduce significant distortion to the original signal.

Among more sophisticated methods that have been proposed to tackle this problem, various implementations of adaptive filters have been shown as highly effective (Thakor and Yi-Sheng 1991, Lee and Lee 2005, Romero, Geng and Berset 2012). Such filters extract the noise by utilizing a measured reference input which is correlated with the motion artefacts. This is enabled by the fact that heart rate is fairly stable, predictable signal, with the main variable component of the HR signal being the duration between consecutive R-R peaks. This variable interval is influenced by the heart rate changes which result from any activity or other external factors. It poses a problem for the adaptive filter to continuously adjust the impulse responses of the adaptive filter in accordance with the R-waves.

To overcome this problem the proposed adaptive filter uses the accelerometer signal as a noise reference. The accelerometer signal is selected because, on one hand, it is correlated with the motion artefacts present in the noisy ECG signal, while on the other hand, it is highly uncorrelated with the ECG signal. For the purpose of this experiment the SHIMMER node, presented in Figure 5.17 measures both the 3-lead ECG signal and 3-axis accelerometer signal from the chest. The first one constitutes the primary input $d[n]$, while simultaneously measured orthogonal accelerometer signals a_x, a_y, a_z allows to calculate the body motion signal, which is the overall force (net force) acting on the body. It is used as the noise reference input $x[n]$.
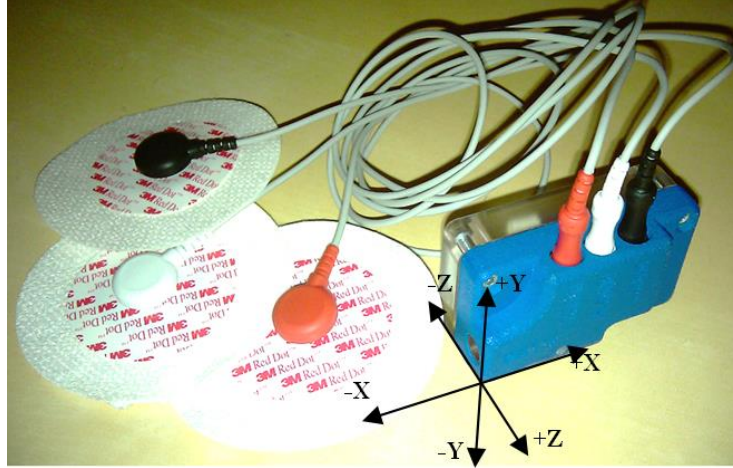
Figure 5.17 SHIMMER node with 3-axis accelerometer coordinates and ECG daughter card.

The noise reference signal is obtained from 3-axial accelerometer which measures the sum of the inertial acceleration components and gravitational acceleration components acting along its axis. It generates an analog voltage that represents an acceleration force (in g units) which is quantized by 12-bit A/D converter. As a result a 3-dimensional acceleration vector is obtained from a tri-axial accelerometer that is formed by three orthogonal measuring axes. Calibration of the accelerometer is based on the work done by Ferraris et al. (Ferraris, Grimaldi and Parvis 1995) and was performed prior to measurements. The acceleration vector measured by a tri-axial accelerometer written in array format is given by equation (24).

$$\vec{a} = \begin{vmatrix} a_x \\ a_y \\ a_z \end{vmatrix} = \begin{vmatrix} a_I \cos(\theta_x) + g \cos(\varphi_x) \\ a_I \cos(\theta_y) + g \cos(\varphi_y) \\ a_I \cos(\theta_z) + g \cos(\varphi_z) \end{vmatrix} \tag{24}$$

where a is the magnitude of the acceleration along the measuring axis, a_I is the magnitude of the inertial acceleration vector, θ is the inclination of the measuring axis with respect to the inertial acceleration vector, g is the gravitational acceleration vector, and φ is the inclination of the measuring axis with respect to the vertical axis (gravity vector). Converted data is then normalised to the range of [-10, 10] and is exposed to a moving average filter with a sliding window of size 4 in order to eliminate high frequency spectra.

Finally the body motion signal $x[n]$, which is the magnitude of the net force acting on the object, can be computed with the Euclidean norm using equation (25),

$$\|a\| = \sqrt{a_x^2 + a_y^2 + a_z^2} \qquad (25)$$

which is a consequence of the Pythagorean theorem since the basis vectors a_x, a_y, a_z are orthogonal unit vectors. Figure 5.18 illustrates the net force on two sectional views at X-Y and Y-Z planes of the accelerometer coordinates system.
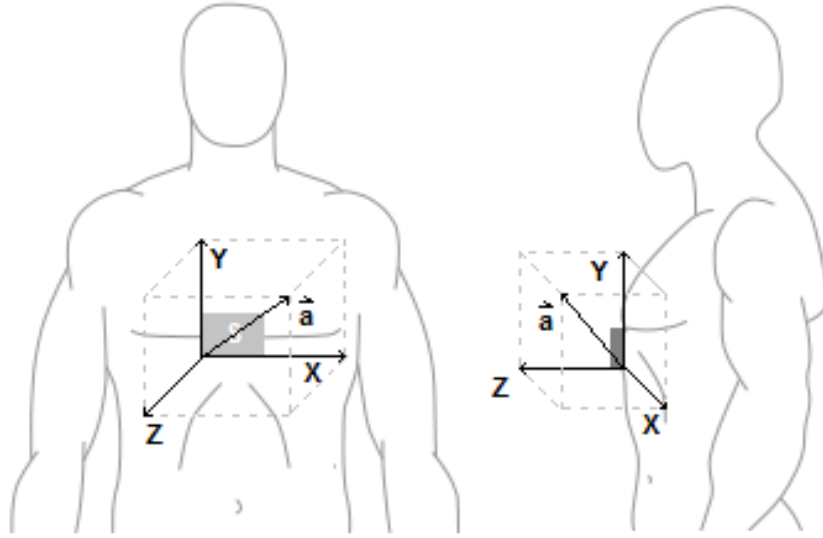
Figure 5.18 Sectional view at X-Y and Y-Z planes of the accelerometer coordinates system.

To test the motion artefacts suppression capabilities of the adaptive filter, the 10-th order FIR filter together with NLMS coefficient adaptation algorithm was implemented and tested on 6 ECG records collected using SHIMMER ECG monitor from two healthy subjects during stand-up/sit-down, walk and run activities. Each subject was asked to perform stand-up/sit-down twice as well as walk and run (jogging) over the period of 10 seconds, in order to keep their heart rate at constant level. This was essential in order to calculate the SNR for which the resting ECG signal was recorded for each subject individually, in order to obtain the subject specific beat-to-beat and QRS morphology. Such obtained morphology along with heart rate measured before and after the exercise, was used to estimate the noise free reference ECG signal. Figure 5.19 shows the reference signal, noisy ECG signal and filtered ECG signal excerpts from stand-up/sit-down record.

ECG signal & Motion artefacts
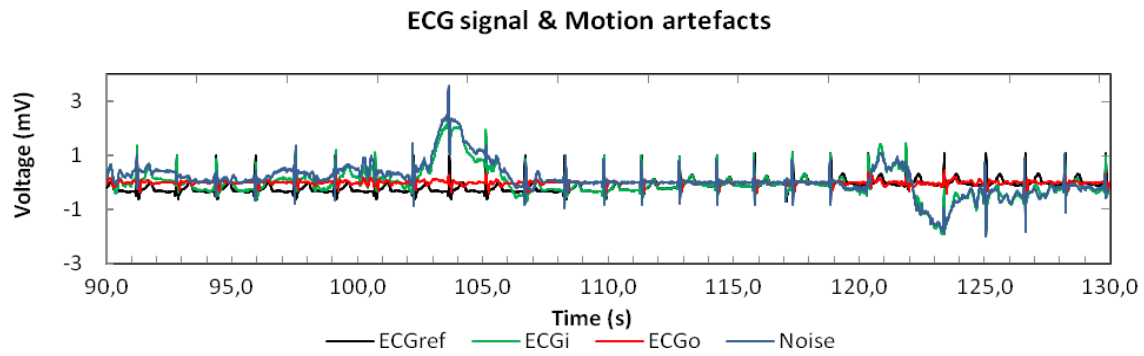
Figure 5.19 Forty seconds ECG signal with motion artefacts (excerpts from stand-up/sit-down record).

The signal-to-noise ratios for all representative activities for both test subjects before and after filtering are presented in Table 5.4. It should be noted that the signal analysed in the table below, was not filtered by any of the previously discussed FIR filters, what explains the negative SNR of the signal after filtration. On average, after the application of NLMS adaptive filter, the SNR of the recorded signal increased by 5.3 dB. The best to satisfactory artefact reduction performance was achieved with low to moderate intensity activities such as stand-up/sit-down and low pace walking for which the SNR increased by 8.2 dB and 4.6 dB respectively. The worst artefact reduction performance can be attributed to high intensity activities such as run, for which largely negative SNR increased by 3.1 dB.

RMS	stand-up/sit-down				walk				run			
	Subject 1		Subject 2		Subject 1		Subject 2		Subject 1		Subject 2	
ECG_{ref}	0.101	0.105	0.327	0.325	0.322	0.323	0.107	0.107	0.107	0.104	0.104	0.100
ECG_i	0.358	0.393	0.793	1.480	0.542	0.648	0.220	0.219	0.427	0.456	0.212	0.305
ECG_o	0.109	0.153	0.301	0.369	0.159	0.256	0.114	0.082	0.333	0.498	0.142	0.192
$ECG_{ref} - ECG_i$	0.345	0.384	0.912	1.582	0.565	0.686	0.208	0.201	0.435	0.458	0.232	0.299
$ECG_{ref} - ECG_o$	0.108	0.165	0.492	0.495	0.346	0.381	0.135	0.110	0.348	0.275	0.170	0.201
SNR [dB]												
SNR_{before}	-10.7	-11.3	-8.9	-13.8	-4.9	-6.6	-5.7	-5.5	-12.2	-12.9	-7.0	-9.5
SNR_{after}	-0.6	-4.0	-3.6	-3.7	-0.6	-1.5	-2.0	-0.2	-10.2	-8.4	-4.3	-6.0
$SNR_{increase}$	10.1	7.3	5.4	10.1	4.3	5.1	3.7	5.2	1.9	4.4	2.7	3.4
$Avg(SNR_{increase})$	8.2				4.6				3.1			
	5.3											

Table 5.4 Signal-to-Noise ratios for the motion artefacts noise contaminated ECG signal before and after the application of the adaptive filter.

It should be noted that even though this method increases the SNR under running conditions, since the total amount of noise is much larger than other conditions, the running ECG after artefact reduction is still quite noisy with negative SNR. Under running conditions the electrical potentials, that affects the ECG signal, are not only the result of the chest motion but also tremors coming from other part of the body. The conclusion can therefore be drawn that the proposed method is less effective under high intensity activities compared with low to moderate intensity activities, mostly correlated with chest movements, for which the large portion of noise can be removed as presented in Figure 5.20 below. With regards to the vital signs monitoring application proposed in this work, this method gives satisfactory result as the high intensity activities won't be subject for analysis by the presented model.

134
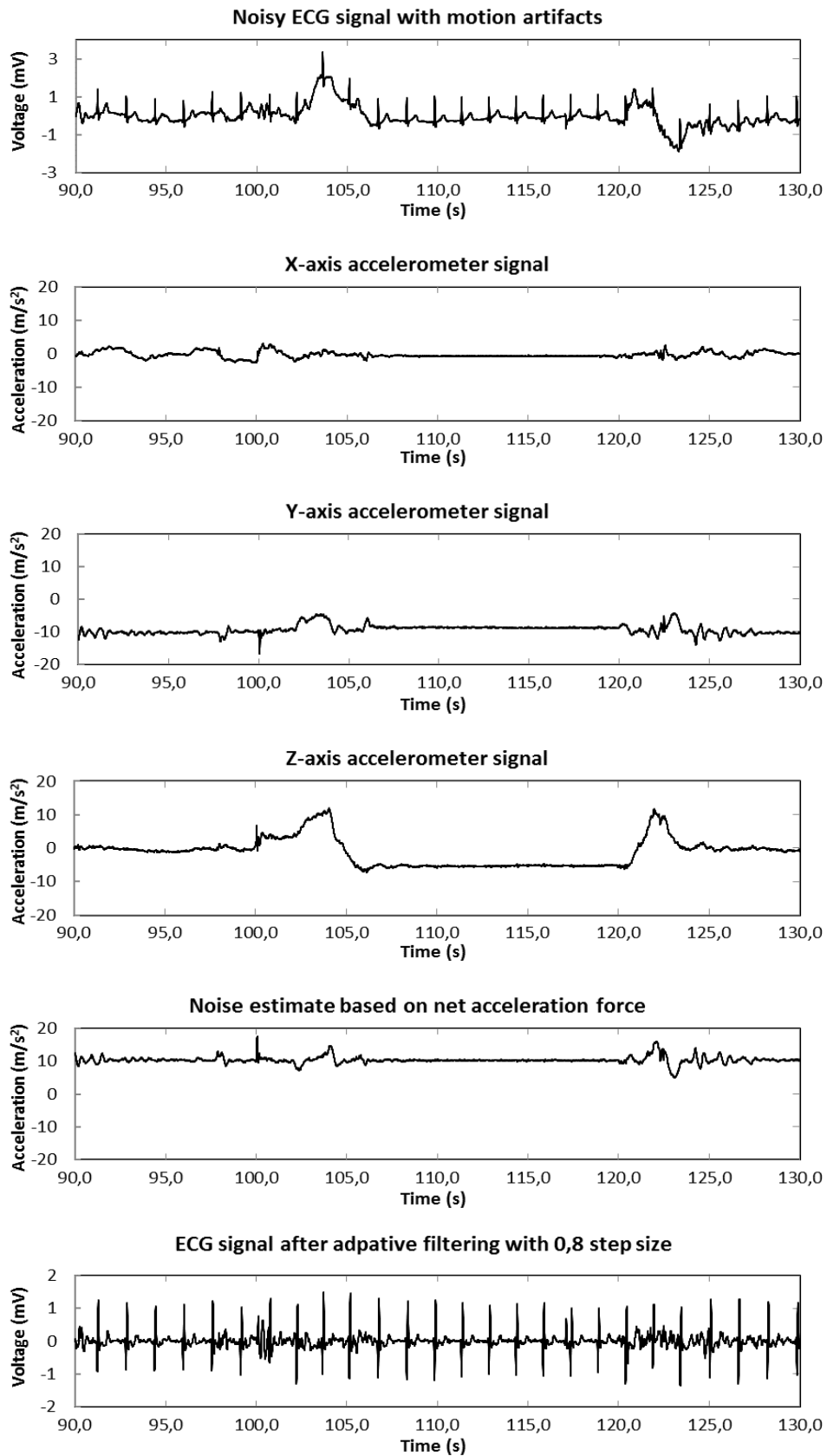
Figure 5.20 Forty seconds excerpt from SHIMMER AccelECG sensor node record, collected during stand-up/sit-down activities, before and after the accelerometer based adaptive filtering.

The Figure 5.21 below is the block diagram for the complete ECG monitoring system. The final noise-free ECG signal is located between 0.5 Hz and 25, Hz depending on the individual. Application of peak detector to such obtained signal, which is presented in chapter 7, allows to calculate average peak-to-peak intervals that are used to calculate heart rate.
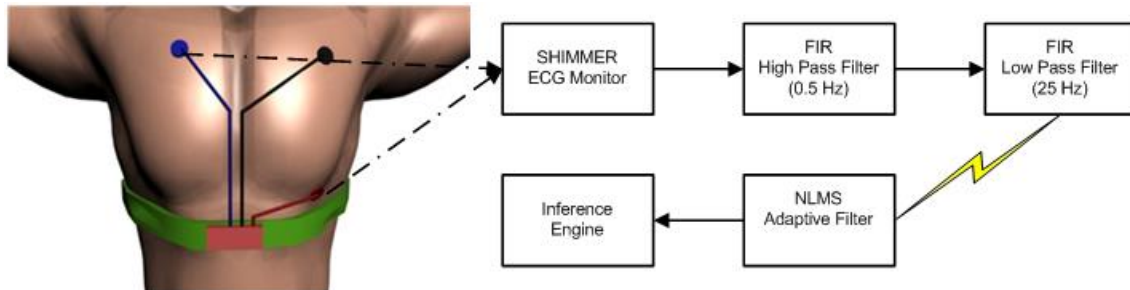
Figure 5.21 Block diagram of the ECG signal processing.

5.3.2   Respiratory rate monitor

As described in chapter 2 the simplest respiratory rate measurement method base on chest and abdomen observation. This is possible due to chest movements associated with each inspiration and expiration. Such secondary effect of breathing can be utilised to build a simple respiratory rate monitor. In its principles, such monitor base on piezoelectric effect, which accumulates some charge in certain solid materials, notably crystals, in response to applied mechanical strain. The piezoelectric sensor attached to the subject' chest, can generate an electric potential in response to a mechanical stress applied to an elastic material band, which holds the sensor.

Although piezoelectric sensor is sufficient to give us a raw, sinusoidal signal, such signal needs further processing. First of all, the signal is prone to contamination by noise, caused by the body moves that are not related to breathing process. Additionally, due to the physiological nature of the signal from the human body the signal amplitude is extremely small - approximately between -1 to 1 mV for tidal volume exchange and between -5 to 10 mV for expiratory and inspiratory reserve volume exchange. The amplitude, however, may vary due to the stress applied to sensor as well as other factors specific to piezoelectric material used. Therefore, in order to measure this signal with the analog-to-digital converter (ADC), the signal needs to be amplified first. This will allow to reserve some margin for any variations between subject due to amount of stress applied on elastic band.

The filtering and amplification problem can be addressed with two post processing blocks: a low pass filter to remove the motion noise and an amplifier to increase the signal amplitude to the ADC quantifiable value. The purpose of the filter is to keep any frequency below 1 Hz due to respiratory rate characteristics (see respiratory rate ranges in chapter 2) and eliminate everything above this range. The purpose of the amplification stage is to amplify the remaining signal as close to reference voltage of 5V. The figure 5.22 below presents the block diagram for the complete system.
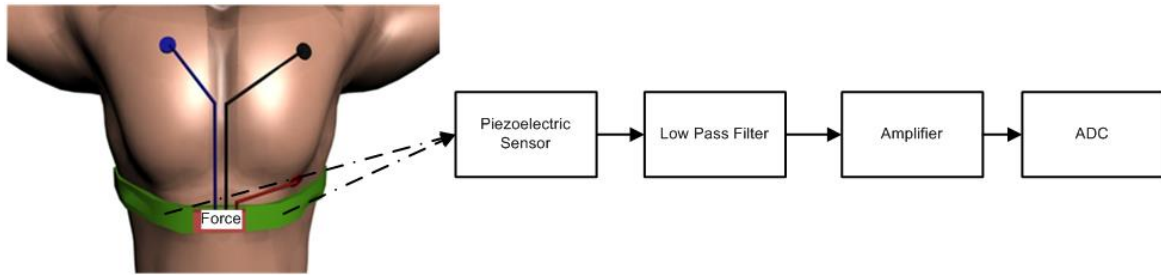
Figure 5.22 Block diagram of the respiratory rate signal processing.

The block diagram of the respiratory rate signal processing system is made of three circuits: piezoelectric sensor circuit, analogue low pass filter and amplifier. Each component implementation is depicted and presented on schematic diagrams below. The assembled respiratory rate monitor circuit together with a complete printed circuit board design, used in the implementation, is presented next.

The first element of the circuit, responsible for respiration force signal generation, is presented in the schematic diagram in Figure 5.23. It is made of the piezoelectric sensor (crystal) and a voltage offset, which are connected in series. Due to constriction and expansion of the chest the piezoelectric sensor generates bipolar sinusoidal signal. Because the whole circuit is powered from a single +5V DC voltage source, in order to be able to amplify this signal the respiratory rate signal were superimposed by the DC level, so the initial potential range of (-5mV;10mV), generated by the piezoelectric sensors, has been shifted to (20mV;35mV). It has been achieved by implementing a voltage shifter in form of potentiometer deployed as fixed resistors power voltage divider. Such superimposed signal oscillates around 25mV rather than a 0V.
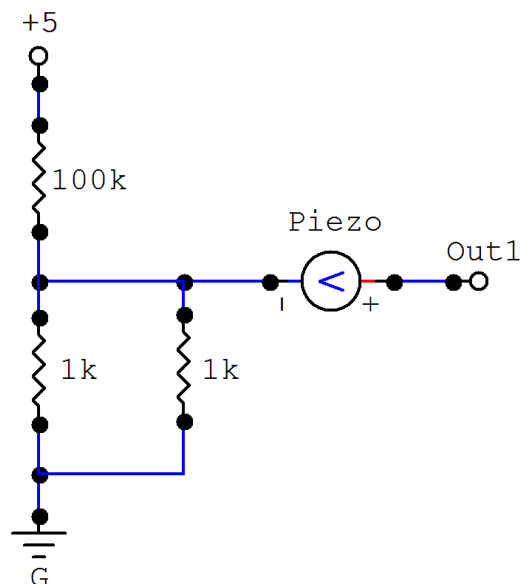
Figure 5.23 Schematic diagram of the piezoelectric sensor and a 25mV voltage offset circuit.

The prototype use a SleepSense 1370 Piezo Effort Sensor (S.L.P. Inc. 2010). It converts chest or abdominal respiration movement using a piezoelectric crystal to a small analogue

voltage that provides a clear, reliable indication of respiration waveforms. The generated signal is typically 1mV peak to peak however its amplitudes may vary. Response is dependent on such variables as sensor position, strap tension and polygraph characteristics, etc. Its isolated connectors and wires provide excellent immunity to environmental artefacts, therefore the signal is very strong and stable.

Figure 5.24 SleepSense 1370 Piezo Effort Sensor.

Obtained signal undergoes further processing by the passive low pass filter in order to remove the excess noise. Schematic diagram of this block has been presented in Figure 5.25. The purpose of the filter is to keep any frequency response below 1Hz and eliminate all frequencies above. It is accomplished via the combination of the 160kΩ resistor and the 1µF capacitor.
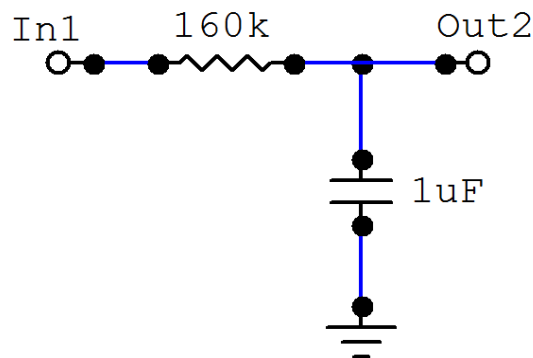
Figure 5.25 Schematic diagram of the low pass filter.

Finally noise free signal is a subject to amplifier operation. Schematic diagram of this block has been presented in Figure 5.26. The purpose of the amplification stage is to amplify the remaining signal as close to 5V as possible, leaving some margin for the varying signals amplitude. Such amplified signal is further presented at the analog-to-digital converter (ADC) inputs. This is accomplished by using an LM358N OPAMP in non-inverting amplifier configuration with the combination of a 100kΩ resistor and a 1kΩ resistor. This gives us a gain of approximately 111x amplification.

Figure 5.26 Schematic diagram of the amplifier.

The complete schematic diagram of the respiratory rate monitor is shown in Figure 5.27. The voltage output from this circuit is connected to channel A7 of the SHIMMER ADC, and represents the force applied to piezoelectric sensor.
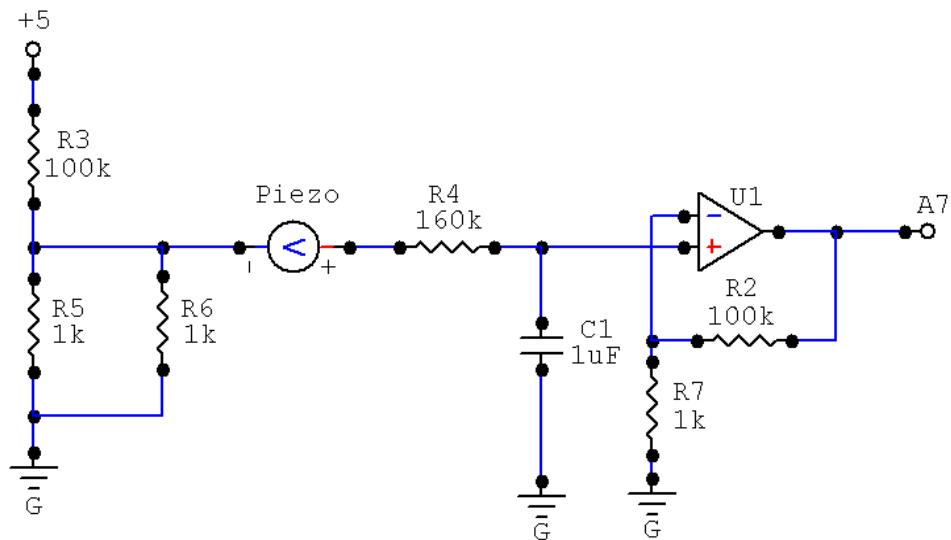
Figure 5.27 Schematic diagram for the complete respiratory rate monitor circuit.

5.3.3   Temperature monitor

To implement the body temperature monitor, the MA100 thermistor from GE Industrial Sensing (Ge Industrial Sensing 2006) has been used. It is a NTC type biomedical chip thermistor exclusively design for biomedical applications. It fulfils the requirements of such applications, in terms of sensitivity, which ranges from 0ºC to 50ºC and its compact size of 0.76 x 9.52 mm. MA100 chip thermistor assemblies are designed for use in applications involving both intermittent and continuous patient temperature monitoring. These highly stable, precision thermistor provides fast response times along with reliable measurement due to the tight interchangeable tolerance for the temperature range of 35 to 39 °C falls to ±0.05 °C. To get accurate auxiliary temperature readings, the temperature sensor is sewn into the elastic chest band and placed in direct contact with skin.

5.3.3.1   Temperature monitor circuit design

A negative temperature coefficients (NTC) thermistor is a two terminal solid state electronic component that exhibits a large, predictable change in resistance corresponding to changes in absolute body temperature. This change in body temperature of the thermistor can be brought about either externally via a change in ambient temperature or internally by heat resulting from current passing through the device or by a combination of these effects. It measures the temperature by representing it as voltage values, which are then quantised by the analogue-to-digital converter. The ideal thermistor generates the voltage output that has a linear correlation with the temperature. However, in reality the linearity of this relationship is the approximation which is valid for a specific short temperature ranges only.

The solution to this problem is to implement linearisation with the analogue hardware. The simple approach, presented in Figure 5.28, places the thermistor in series with the standard resistor and a voltage source.
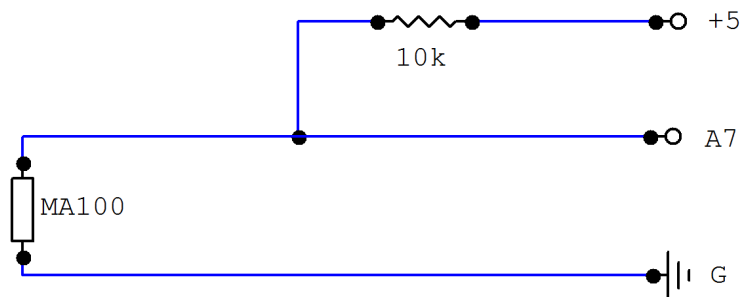
Figure 5.28 Schematic diagram of the MA100 thermocouple.

The temperature response and linearity of these system is shown in figure 5.29. In this figure, the series thermistor system responds to temperature in a linear manner over a limited temperature range. The linearisation resistor's value was set to 10kΩ, what is equal to magnitude of the thermistor at the mid-point of the temperature range of interest. This creates a response, where the output slope of the resistive network is at its steepest at this mid-point temperature. If high precision is required, this range is typically +/-25°C around the nominal temperature of the thermistor at the 10kΩ value.
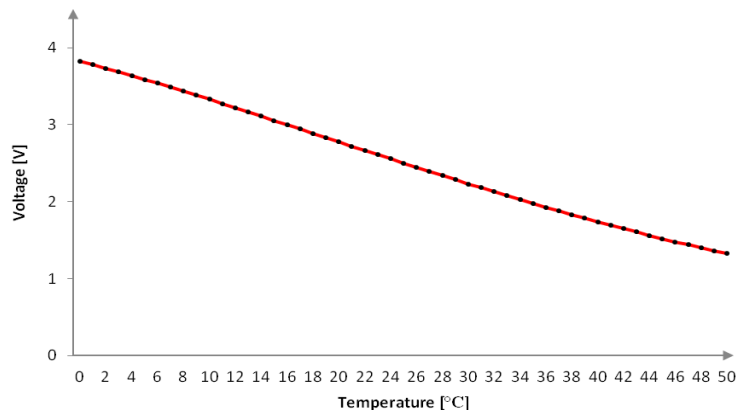
Figure 5.29 Temperature response of the thermocouple circuit.

In order to translated such obtained voltage measurement to a temperature in degree Celsius the equation (26) is used.

$$TEMP = \frac{V_{0°C}[mV] - V_{out}}{\Delta V[mV/°C]}$$

(26)

where:

$TEMP$	corresponds to the temperature measurement, in degree Celsius (°C).
$V_{0°C}$	corresponds to voltage measurement (in mV) at 0°C temperature
V_{out}	corresponds to the measured voltage at the terminals of the MA100 thermistor
ΔV	the constant value in mV, which corresponds to the variation of increasing temperature value by 1 degree Celsius (°C).

ΔV in the above equation has been obtained experimentally for the circuit presented in figure 5.28. It represents thermistor's ability to measure temperature change. It can be expressed as the voltage response on increasing temperature value by 1 degree Celsius (°C) at either a single temperature point or over a temperature range. Figure 5.30 presents such temperature response for our circuit over the entire temperature range of interest. Its average value of 50.08 mV is used as ΔV constant in the equation (34).
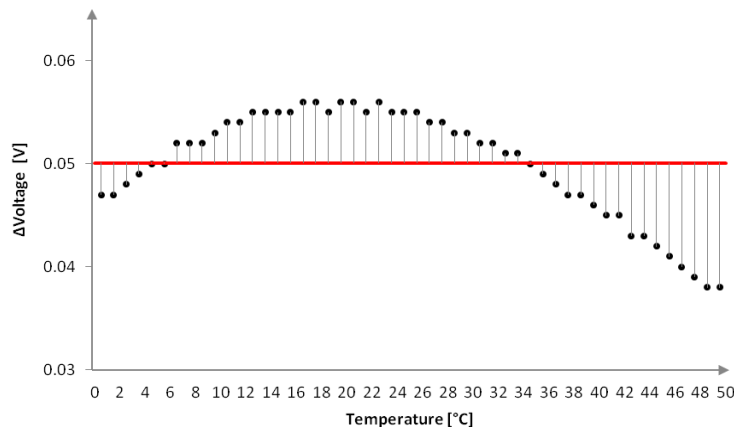
Figure 5.30 Voltage response on increasing temperature value by 1 degree Celsius (°C).

5.3.4 Implementation

The following Figure 5.31 depicts the complete printed circuit board design, used for the implementation. The components naming convention used in this figure corresponds to the schematic diagrams, presented in figure 5.27. and 5.28. The difference between those two diagrams is the power supply for the operational amplifier, which is not shown in the above schematics for the sake of clarity. Additionally, both MA100 temperature sensor and piezoelectric effort sensor are not hard wired to the board, but connects to the J1 header strip connector as follows: T – temperature sensor, TG – temperature sensor ground, R – respiratory

sensor, RG – respiratory sensor ground. In actual prototype implementation we have used the following off-the-shelf components: 7x typical axial-lead resistors of various resistances, 1x ceramic capacitors, 1x LM358N operational amplifier from Texas Instruments, and 2x header strip connectors of size 4x1.
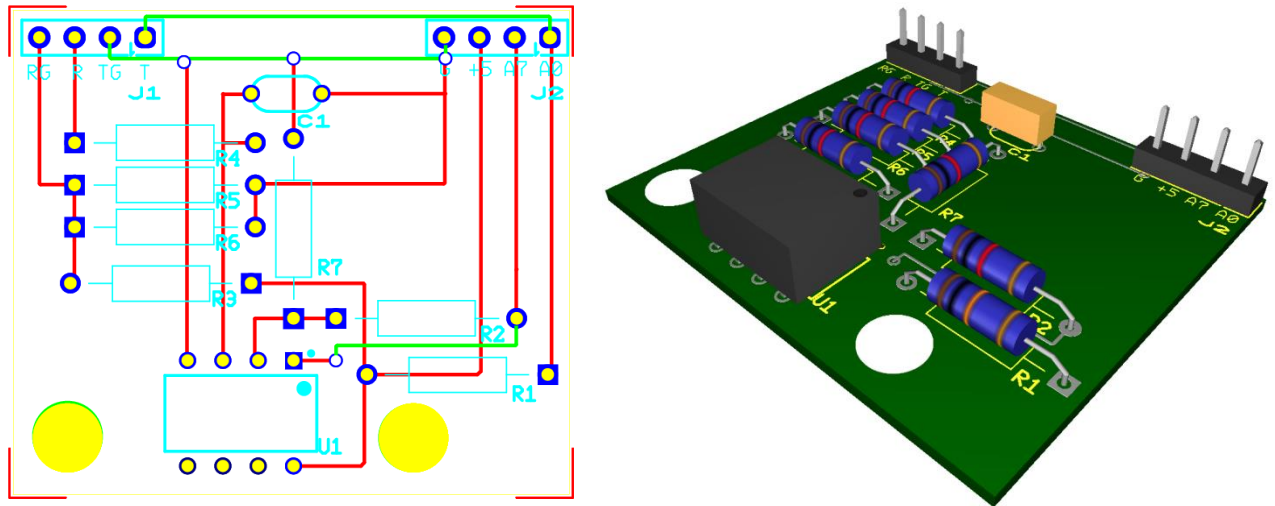
Figure 5.31 Respiratory rate and temperature monitor PCB design.

5.4 Armband

Armband sensor node design presented below focus on design of a pulse oximeter, which builds upon design of (Townsend 2001) and (Qureshi 2009). Prototype armband is made of an elastic material, which holds the sensor and twist around the wrist. The pulse oximeter uses two red and infrared LED / photo-detector pairs, which detect the wavelength of the emitted red and IR light, measuring in such a way the oxygen saturation of blood flowing through the subject's fingertip or earlobe.

5.4.1 Pulse oximeter

As described in chapter 2, pulse oximetry is the measurement of light transmittance/ absorbance through the finger or earlobe. Therefore to build a simple pulse oximeter, the light emitter (LED circuit) and a light detector (photodetector circuit) is used, in order to convert transmitted light into an electrical signal. The red and infrared wavelengths, yield the best results having the greatest separation between the haemoglobin and oxyhaemoglobin absorption spectra. However due to two wavelengths, there have to be two separate light emitter an detector circuits. Such detected electrical signals are currents, which correspond to the intensity of light. The simple rule is that the higher the current is the greater the light intensity.

According to the specification of Analog-to-Digital converter (ADC), which is present on board of MSP430 CPU, more useful value of interest for us is the voltage rather than a current.

Therefore, we will need to convert obtained currents into corresponding voltages, what can be achieved via the current-to-voltage convertor made of a single operational amplifier (op-amp). Although these elements are sufficient to give us a raw, pulsatile signal, it requires further processing to extract high fidelity oxygen saturation values. Initial analysis of the obtained signal revealed some signal characteristics, which make such signal impractical without further processing.

Firstly, the pulsatile signal is superimposed on a much larger DC offset, caused by the fact that the light illuminates both arterial and venous blood and must traverse all tissues between light source and receiver, what was discussed in section 3.6.2. The measured DC offset is approximately 100 times that of the amplitude of the AC signal. Secondly, it carries an overwhelming amount of noise that comes from various sources. These source are: the 60 Hz power line noise, motion noise from user movements, and noise from any surrounding light sources (the detecting from the visible spectrum as well. Finally, due to the physiological nature of the signal from the human body the AC signal amplitude is extremely small approximately 1-10 mV. This becomes a problem when the signal needs to be post processed. Therefore the signal needs to be amplified so that the analog-to-digital converter (ADC) can actually detect and quantise it.

Issues mentioned above can be addressed with two post processing blocks: a band pass filter to remove the excess noise and DC offset as well as an amplifier to increase the signal amplitude to our desired value. The purpose of the filter is to keep any frequency content between 0-5Hz and eliminate any frequency above and below this range. The purpose of the amplification stage is to amplify the remaining signal as close to 5V as possible.

The figure 5.32 below presents the block diagram for the complete pulse oximetry system. It includes LED - photodiode pair, current-to-voltage converter and post processing blocks in order to deal with signal conditioning as mentioned above. There are two of these systems, one for the red and one for the infrared LED.
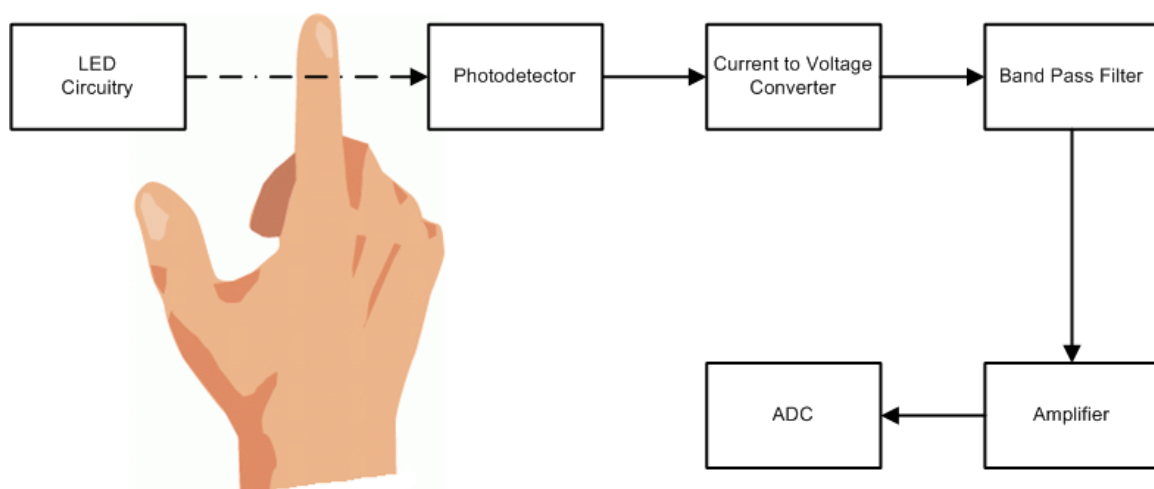
Figure 5.32 Block diagram of the pulse oximeter signal processing.

143

5.4.1.1   Pulse oximeter circuit design

The system described above can now be depicted and presented on schematic diagrams for each block. Firstly, each block will be presented separately, with detail description of components used to build it. The whole pulse oximeter circuit, together with a complete printed circuit board design, used for implementation, will be presented next.

The simplest realisation of pulse oximeter, presented in schematic diagram in Figure 5.33, can be achieved with the LED – photodetector pair together with current-to-voltage converter circuit. In order to be able to separate the haemoglobin and oxyhaemoglobin the pulse oximeter system requires two of such circuits, one for red and one for infrared light. With 5 V power supply for these circuit and additional 100 Ω resistor is needed. It regulates the current across the LED according to its specification. This ensures that the current flow through the LED, will allow to produce the light at adequate wavelength, that after passing through the finger, could be successfully detected by the corresponding photodetector. The approximate current through this system is 50 mA, assuming the LED resistance is negligible.

The transmitted light through the finger is detected and represented as current by the photodetector, implemented as a photodiode. The incident light transmitted through the finger strikes the photodiode causing current to flow. This current then flows through the 2MΩ resistor and creates a voltage at the output port of the operational amplifiers (op-amp) what consitutes the signal of interest. This unit is called current-to-voltage converter and realize the function of current to voltage conversion. Schematic diagram presented below is the simplest pulse oximetry monitor.
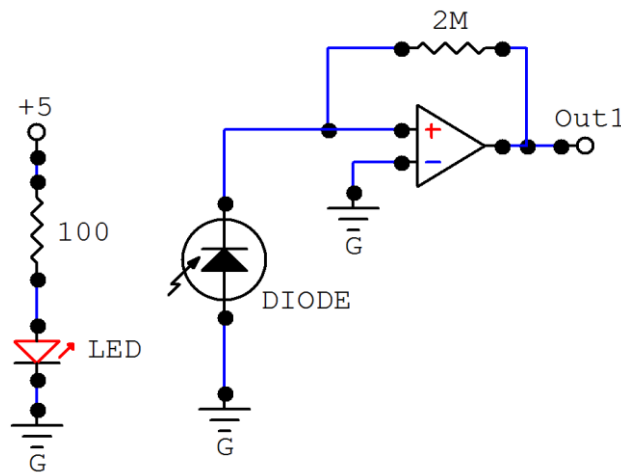
Figure 5.33 Schematic diagram for the LED – photodiode pair and a current-to-voltage converter blocks.

In the actual implementation as a LED-photodiode pair we have used a Nonin 8000J Adult Flex Sensor (Nonin Medical Inc 2010). It combines the convenience and performance of a disposable sensor with the economics of a reusable sensor. The reusable Flex Sensor, made of a durable silicone, is secured with a matching disposable FlexiWrap adhesive as seen in Figure 5.34. This hybrid sensor system is ideal for patients weight greater than 20kg and is suitable for numerous applications including extended monitoring and motion situations.

Nonin 8000J Adult Flex Sensor use PureLight™ technology, which produce a high-intensity pure light spectrum, which eliminates variations in readings from patient-to-patient and sensor-to-sensor. The red LED emits a light of 660nm wavelength, which measures deoxygenated haemoglobin while the infrared LED emits a light of 910nm wavelength, which measures oxygenated haemoglobin. Use of PureLight™ LEDs do not cause a shift in the oximeter's calibration curve at SpO_2 levels below 80%, as opposed to common red LEDs that may produce a secondary spectrum emission, called contamination, what impacts the oxygenated haemoglobin measurement and corrupt readings. Nonin 8000J Adult Flex Sensor interface through D-sub DE-9 pin connector.
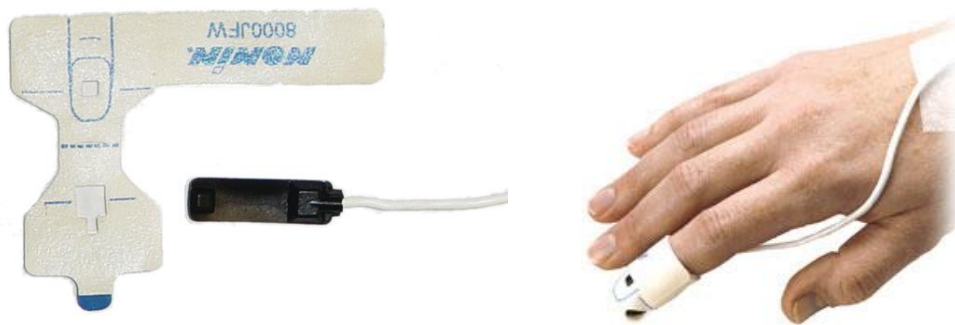
Figure 5.34 Nonin 8000J Adult Flex Sensor and disposable FlexiWrap adhesive.

Obtained signal undergoes further processing by the passive band pass filter in order to remove the excess noise and DC offset. Schematic diagram of this block has been presented in Figure 5.35. The purpose of this filter is to keep any frequency content between 0-5Hz and eliminate any frequency above and below this range. The low pass cut-off for this band pass filter is set to approximately 5Hz, what is accomplished via the combination of the 330kΩ resistor and the 0.1μF capacitor. The high pass cut-off of for this filter is set to approximately 0.5 Hz and is accomplished via the combination of the 3MΩ and the 0.1μF capacitor. Additionally the filter has a pre-amplification stage implemented as the combination of 10kΩ and 1kΩ resistors at the inverting input of the operational amplifier. This combination gives approximately 11x amplitude gain prior to sending the signal to the amplification stage.
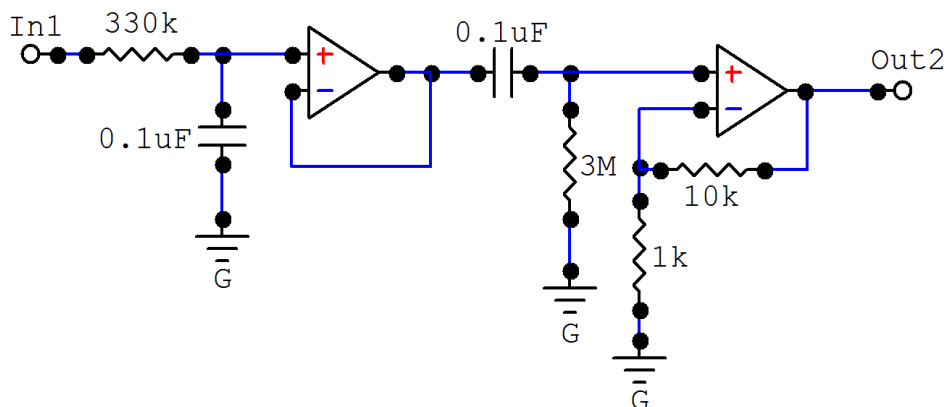
Figure 5.35 Schematic diagram for the band pass filter with a small pre-amplification gain.

Finally noise and DC offset free signal is a subject to amplifier operation. Schematic diagram of this block has been presented in Figure 5.36. The purpose of the amplification stage is to amplify the remaining signal as close as possible to 5V, so that a detectable signal could be presented at the inputs of analog-to-digital converter (ADC). This is accomplished by using an LM324N OPAMP in non-inverting amplifier configuration with the combination of a 56kΩ resistor and a 1kΩ resistor. This gives us a gain of approximately 57x amplification with respect to original signal.
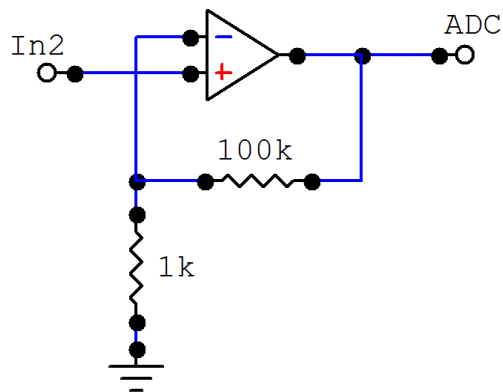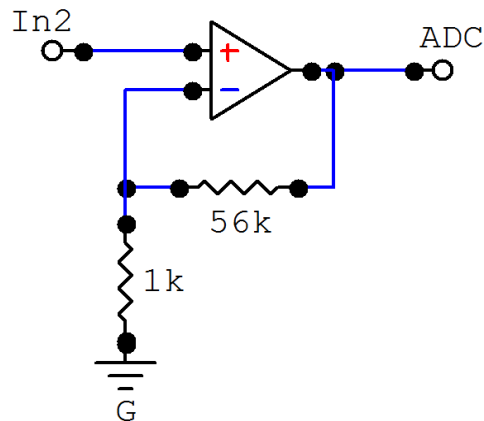

Figure 5.36 Schematic diagram for the amplifier.

The complete schematic diagram for the whole (red & IR light) pulse oximeter is presented in the Figure 5.37.
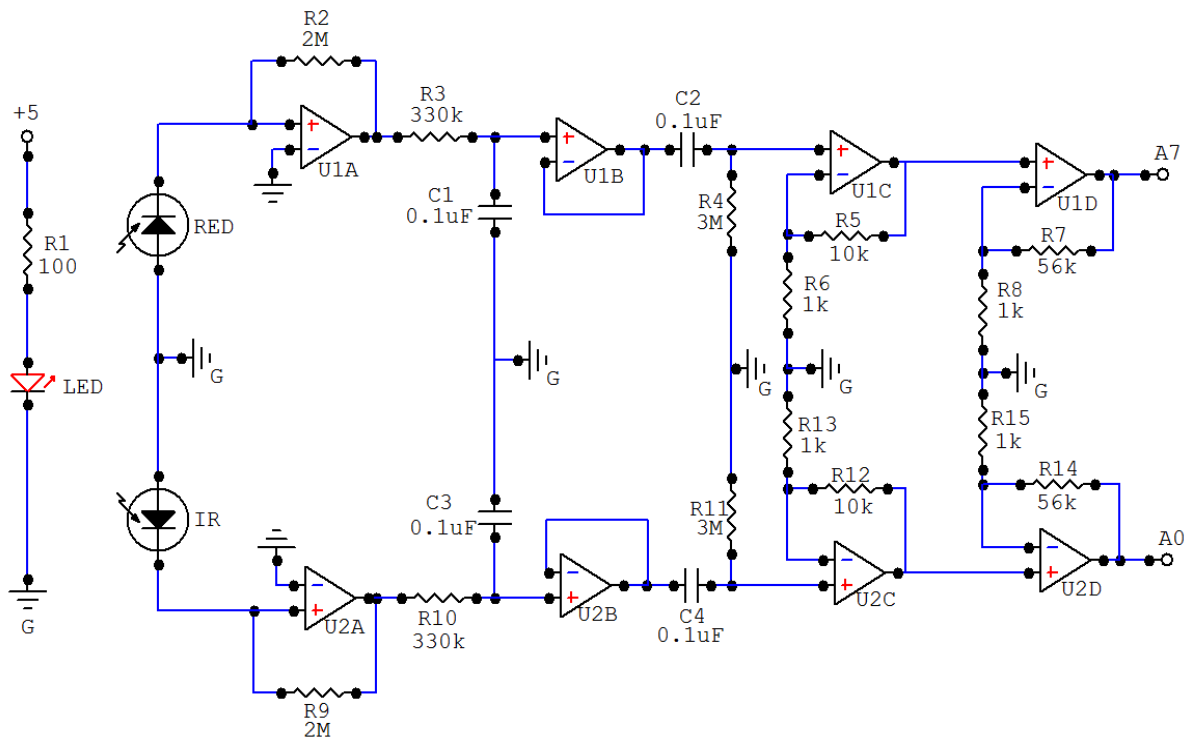
Figure 5.37 Schematic diagram of the pulse oximeter including red and infrared circuits.

5.4.1.2 Calculation of the pulse oximetry

As described in chapter 2, hemoglobin is the oxygen carrying molecule of the blood. It exist in two forms: oxygenated hemoglobin denoted HbO_2 and deoxygenated hemoglobin denoted Hb. Oxygen saturation denoted as SpO_2 refers to the ratio of oxygenated hemoglobin to the total concentration of hemoglobin (Townsend 2001) as defined by equation (27).

$$SpO_2 = \frac{HbO_2}{total\ hemoglobin}$$ (27)

If we assume initially that the transmission of light through the artery is influenced only by the relative concentrations of oxygenated and deoxygenated hemoglobin and their absorption coefficients at the two measurement wavelengths, then the light intensity will decrease logarithmically with path length according to the Beer−Lambert law, which relates the absorption of light to the properties of the material the light is passing through (Townsend 2001). With such assumption, simply a difference in light absorption spectra of HbO2 and Hb can be used. If we consider an artery of length I through which light, initially of intensity I_{in} is passed, this law states that:

At wavelength λ_1 $I_1 = I_{in1} 10^{-(\alpha_{o1}C_o + \alpha_{r1}C_r)l}$

At wavelength λ_2 $I_2 = I_{in2} 10^{-(\alpha_{o2}C_o + \alpha_{r2}C_r)l}$ (28)

where:

- C_o is the concentration of oxyhaemoglobin (HbO2)
- C_r is the concentration of reduced haemoglobin (Hb)
- α_{on} is the absorption coefficient of HbO2 at wavelength λ_n
- α_{rn} is the absorption coefficient of Hb at wavelength λ_n

Using these principles we can obtain an expression (29) for the ratio of the intensity of light transmitted at two different wavelengths given by (Townsend 2001) as:

$$R = \frac{\log_{10}(I_1/I_{in1})}{\log_{10}(I_2/I_{in2})}$$ (29)

Once we know the absorbance coefficients of HbO2 and Hb at the two wavelengths, we can find the oxygen saturation via the following formula in equation (30):

$$SpO_2 = \frac{C_o}{C_o + C_r} = \frac{\alpha_{r2}R - \alpha_{r1}}{(\alpha_{r2} - \alpha_{o2})R - (\alpha_{r1} - \alpha_{o1})}$$ (30)

In practice however, where the pulse oximetry is measured based on the intensity of light transmitted across, for examples the fingertip or earlobe, the equation (30) does not take into account the attenuation of light by other body segments, through which the light must pass. These are: arterial blood (the one of interest), venous blood and tissues. Such pulsatile signal, which is the result of such measurement varies in time with the heart beat and is superimposed on a direct current (DC) level. The amplitude of this cardiac−synchronous pulsatile signal is approximately 1% of the DC.

If we assume that the increase in attenuation of light is caused only by the inflow of arterial blood into the fingertip, we can calculate the oxygen saturation of the arterial blood by subtracting the DC component of the attenuation from the total attenuation, leaving only the cardiac−synchronous pulsatile component for the dual−wavelength determination of oxygen saturation (Townsend 2001). Therefore, the general oximetry equation (29) derived earlier is equally valid for pulse oximetry if R is now given by more practical equation (31), in terms of light intensity measured as direct current (DC) and alternating current (AC) at λ_1 or λ_2.

$$R = \frac{\log_{10}\left(I_{dc+ac}/I_{dc}\right)_{\lambda_1}}{\log_{10}\left(I_{dc+ac}/I_{dc}\right)_{\lambda_2}} \tag{31}$$

Because as it can be seen from the block diagram in in Figure 5.35, the proposed pulse oximeter device is equipped in the band pass filter, which removes the DC level, hence the DC level can be eliminated from the formula (31). These new index used by the microprocessor to calculate the oxygen saturation is defined in equation (32) as:

$$R = \frac{\log_{10}\left(I_{ac}\right)_{\lambda_1}}{\log_{10}\left(I_{ac}\right)_{\lambda_2}} \tag{32}$$

Once the ratio is calculated, the final SpO2 can be calculated. An approximate formula used to calculating percentage of oxygen saturation, which represents the relationships between the ratio R and the Oxygen Saturation, is defined in equation (33).

$$SpO_2 = 110 - 25R \tag{33}$$

5.4.2 Implementation

The following figure 5.38 depicts the complete printed circuit board design, used in implementation. The red and infrared LEDs are limited to a single light emitter for the sake of clarity. In actual prototype implementation the following off-the-shelf components have been used: 15x typical axial-lead resistors of various resistance, 4x ceramic capacitors, 2x LM324N operational amplifiers from Texas Instruments, 1x D-sub 9 pin female connector (DE-9) and some single pin connectors.
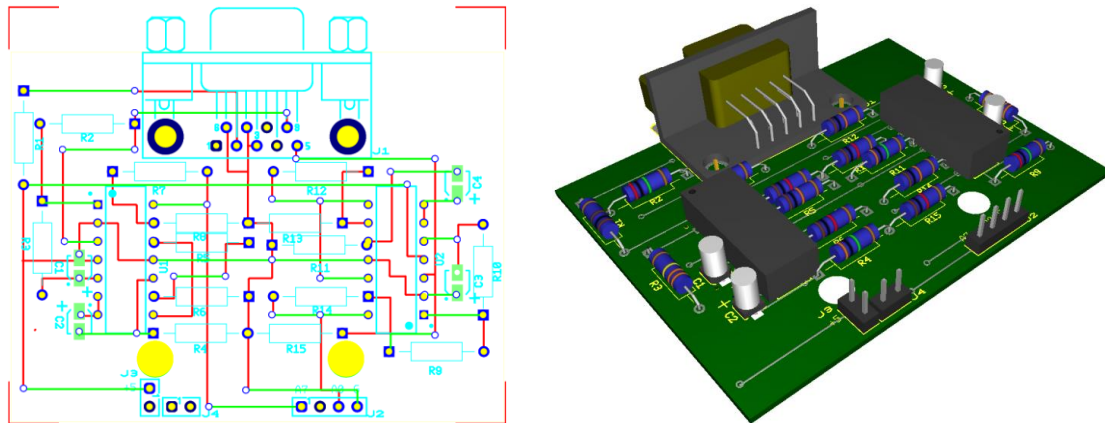
Figure 5.38 Pulse oximeter PCB design.

Chapter

6 Mobile Inference Engine model

6.1 Introduction

This chapter describes both technical and methodological concepts behind the jMENN – joone Mobile Edition Neural network engine based on the open source Java Object Oriented Neural Engine (JOONE) (Marrone 2007). The aim of this chapter is to present the inference engine/runtime environment and the complementary Inference Engine Editor. Moreover the chapter contains a discussion on the logic-centred development and deployment model that this new mobile version of JOONE promotes. Finally, towards the end of this chapter all the efforts and modifications that have been done to the core of the JOONE engine are presented. This changes enabled JOONE's operation on mobile devices, on top of the data aggregation model presented in chapter 4.

6.2 Why Java Object Oriented Neural Engine?

In selection of the inference engine for this project the crucial role played the selection ofg the programming framework which would enable to build AI applications on a local machine, train them on a distributed environment and run on whatever device. Java Framework was the most natural choice which enabled to used desktop, web and distributed environments but what is most important ensured mobile compatibility with today's most popular mobile phone operating system - Android.

The main reason for choosing Java Object Oriented Neural Engine (JOONE) was its lightweight model, making the core engine of JOONE suitable for small devices, having a small footprint and its core engine being runnable on Personal Java environments out-of-the-box. JOONE can be used to build custom systems, adopted in an embedded manner to enhance an existing application, or employed to build applications on mobile devices. Other important criteria was scalability, reliability and expansibility, which JOONE offered through modular architecture based on linkable components that can be extended to build new learning algorithms and neural networks architectures. JOONE's components are pluggable, reusable, and persistent code modules which feature multithreading, serialization and parameterisation. Finally, JOONE components, written by developers, can immediately be used by AI experts and third party designers to build applications by gluing together components with graphical editors,

and controlling the logic with scripts. All this at no extra cost as JOONE is and open source engine licensed under the Lesser General Public License (LGPL), meaning that everyone can freely embed the engine into existing or new applications.

According to Marrone (Marrone 2007) "a long-term goal of JOONE is to become the basic framework to provide a computational engine to AI applications suitable for the mobile devices (phones, PDA, etc.)". Despite of the provision that the package is suitable for small devices running on Personal Java environments, it was unable, without some modifications to its source code, to run the engine on most common mobile device Java profiles: Connected Limited Device Configuration (CLDC 1.1) and the Mobile Information Device Profile (MIDP 2.1). It was mostly dictated by the limitations of this profiles in terms of serialization mechanisms as well as differences between collections, data types and libraries used in Java Standard Edition (Java SE) and Java Mobile Edition (Java ME).

As a result a new package, called joone Mobile Edition Neural Network (jMENN) has been released, as part of this project, what accomplish a goal of this research for mobile inference engine. jMENN is a JOONE engine port to CLDC 1.1 and MIDP 2.1 profiles for use on java Mobile Edition platform. In this chapter a new mobile neural network lifecycle for jMENN is presented, followed by detailed technical discussion on the mobile port, jMENN Editor upgrade and compatibility issues with the JOONE core engine and utilities build for Java Standard Edition platform.

6.3   Neural Network development lifecycle

Neural network development lifecycle consists of four standard phases: building, training, validation and testing phase. jMENN Neural Network Editor supports full development process of mobile neural networks. jMENN Editor is an upgraded version of JOONE Neural Network Editor, a graphical utility tool supplied as part of core engine distribution. Development process is followed by the deployment process and go-live phase which focuses on embedding and use of persisted neural network in java application code. Further in this section a simple process of algorithm development is presented. For more details on development process with Neural Network Editor see (Marrone 2007).

6.3.1   Building phase

Building a neural network with jMENN Neural Network Editor is as simple as connecting consecutive layers of neurons (nodes) with synapses, building more and more complex architectures of any type. Each layer is composed of a certain number of neurons, each of which has the same characteristics (transfer function, learning rate, etc). A neural net can be composed of any number of layers belonging to different typologies (linear, sigmoid, etc.). In this manner any kind of neural networks architecture can be built. Each layer reads its input,

apply the transfer function, and writes the result to its output synapses, to which there are other layers connected and so on. The synapse, in turn, represents the connection between two layers, permitting a pattern to be passed from one layer to another. This is the 'memory' of a neural network, where all weigh of each connection are stored. Hence, it represents a shared resource between two layers, it cannot connect more than two layers.
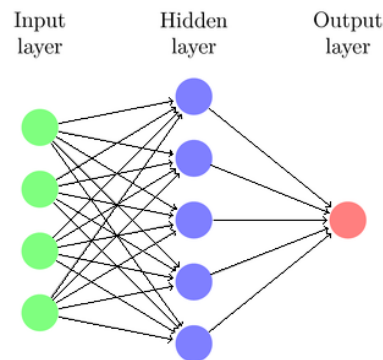
6.1 Example of a simple neural network.

6.3.2   Training phase

When the neural network architecture is build the training phase can begin. In order to control the network training process, manage network events and control how patterns and the internal weights are represented a monitor component must be added. It contains network parameters that the training algorithms use. Other elements that have to be added at this stage are the input and output synapses. This elements make possible the connection of a neural network to external sources of data, either to read the patterns to elaborate, or to store the results of the network to any required data format. Finally, to apply any supervised learning techniques, a neural network needs to have a teacher component that provides the neural network with the error for each input pattern. This is calculated as a difference between the output of the neural network and a desired value obtained from some external data sources. The error is further injected backward into the neural network starting from the output layer of the network, causing each component to process the error pattern and modify the internal connections weight. This process is specific to the applied learning algorithm. Having all this elements connected to our network we can begin the actual training process.
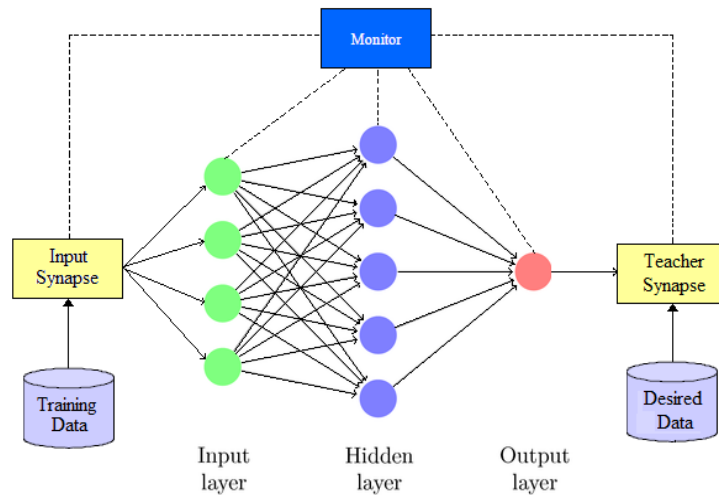
6.2 Simple training ready neural network.

Neural network training begins by presenting the entire training set of data to the network. This set is called epoch. Each epoch consists of multiple specially selected training data patterns that the network has to elaborate. In this process each data instance is being presented one by one to the network, which then calculates its output, calculates the difference between the output and the desired value, propagates this error through the network, adjusts the weights and finally calculates the accuracy over training data.

6.3.3   Validation phase

It is a good rule to reserve a certain number of training patterns to execute the validation check. Validating a neural network during its training cycles is very useful to determine the generalisation capability of the network and to minimise over fitting. Validation is the process of measuring the accuracy over validation data set, that is data set that has not been shown to the network before, or at least the network hasn't trained on it. The validation accuracy is used to control the training process. In general, the training process stops, when the validation accuracy exceeds its accepted threshold. However in the specific case, when the accuracy over the training data set increases, but the accuracy over then validation data set remains the same or decreases, then the overtraining of neural network occurs and the training process should stop. Complete training and validation algorithm is listed below:

```
for each epoch
    for each training data instance
        propagate error through the network
        adjust the weights
        calculate the accuracy over training data
    for each validation data instance
        calculate the accuracy over the validation data
```

```
        if the threshold validation accuracy is met
            exit training
        else
            continue training
```

jMENN engine supports validation mechanism, providing an automatic switch component. It can change its state according to the value of the validation parameter of the Monitor object and connects either to the training or the validation data sets automatically.
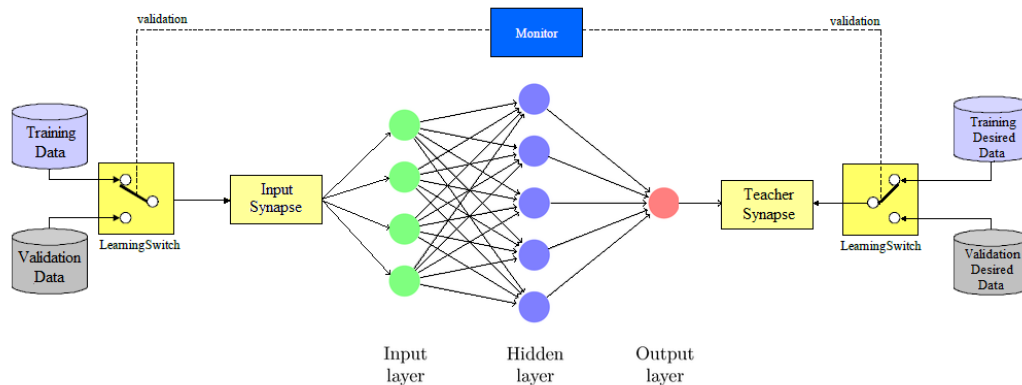
6.3 Neural network validation mechanism.

6.3.4   Testing phase

The last step in neural network development process is to test its performance by calculating the resulting random mean square error (RMSE) for a specific input data set. These patterns cannot be used by the network during the training or validation cycles. Testing begin with the call to `JooneTools.test` method, which accepts as parameters the neural network object containing the network to test, the input test data and the corresponding desired data. The test method returns a double values indicating the RMSE obtained with the input patterns, compared to the given target data. During the testing phase a generalisation capacity of a network on unseen data is calculate.

6.4   Mobile Neural Network deployment process

Deployment process is the most important deference of the Java Mobile Edition neural network lifecycle when compared with the Java Standard Edition lifecycle. Mobile neural network go-live phase, unlike the standard edition one, can only be accomplished embedding the network into a custom mobile application. In standard edition engine version we can also choose to running the network within the Neural Network Editor environment even though it is not a very convenient nor practical form of using it.

Embedding phase takes previously trained, validated and tested network object, which is serialized to a file within editor framework, and deserialize it as part of an existing or new

application or system. Embedding the network hides it from the user, who does not have to know what processing algorithm is used to obtain his results. With a jMENN edition due to internal engine source code modifications this process involves additional converter application. The converter optimizes the collections, data types and libraries used in Java Standard Edition to be compatible with those used in Java Micro Edition. It also replaces not supported `Serialization` mechanism, with custom build `Persistence` mechanism.
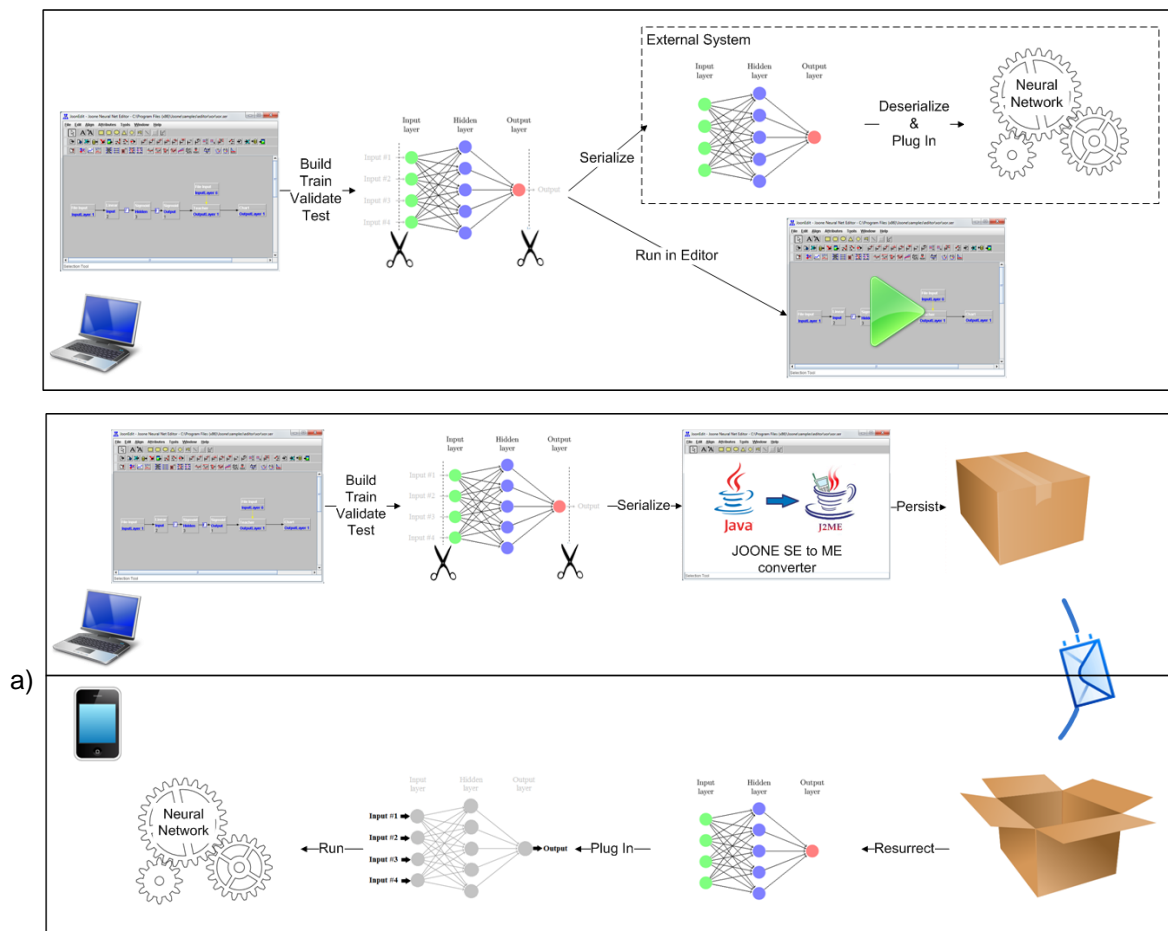
Figure 6.4 Neural Network deployement and use in JOONE a) standard edition, b) mobile edition

jMENNpermits to save the network as a byte stream in the file system, database or transfer it to the remote mobile device using any wired or wireless connection. Difference between the embedding processes with standard and mobile edition is illustrated in Figure 6.4.

6.4.1   Saving and restoring the neural network

In jMENN Mobile Edition saving and restoring the network is more complicated compared to JOONE Standard Edition where all core elements of the engine implement the `Serializable` interface (For more details how to save and restore a network with Java

Standard Edition see (Marrone 2007)). This is because the Connected Limited Device Configuration (CLDC) does not support object serialization or reflection. This means that there is no built-in way to persist objects into a byte stream for CLDC-based profiles.

The work around is to use a custom `Persistence` mechanism. In order to save and restore a network on mobile platform all the core elements of the engine must implement `Persistent` interface. Details on how `Persistence` works can be found in source code modification section. For now let's assume that `Persistent` interface provides `persist` and `resurrect` methods, which corresponds respectively to `writeObject` and `readObject` methods known from `Serializable` interface.

In order to save the entire `NeuralNet` object to mobile edition compatible format, previously build, trained, tested and serialized network needs to be processed by a jMENN converter. It is a custom made plug-in to jMENN version of Neural Network Editor, build as part of this project, which features both `Serialization` and `Persistence` mechanisms, as well as supports necessary type conversions. More details on how the converter application is build can be found in jMENN converter section. For now, to illustrate minor difference in `Persistence` mechanism operation compared to `Serialization` mechanism, to save the network object the converter application use the following code.

```
public static void saveNeuraNet(String fileName) {
try {
      FileOutputStream stream = new FileOutputStream(fileName);
      OutputStream out = new OutputStream(stream);
      byte[] nnetData = nnet.persist();
      out.write(nnetData);
      out.close();
} catch (Exception ex) {
      ex.printStackTrace();} }
```

In contrast to `Serialization` mechanism, which is indirectly called by stream `writeObject` method, `Persistence` requires a direct call to `persist` method of a persisted object in order to recursively save all the objects of a network as an array of bytes. This is then further streamed out to a file using simple `OutputStream`. `Persistence`, the same as `Serialization,` avoids storing the same object's instance twice in case it is referenced by two separated objects.

According to a jMENN development lifecycle, network persisted with jMENN converter is ready to be send and restored on the mobile device. In order to do so the byte stream from where the network was stored: from a file, database, or any other data source have to be read, and a new NeuralNet object using default constructor have to be created. Having such object defined, the entire network can be restored by calling a `resurrect` method. The following listing presents a sample J2ME code that does this.

```
public static NeuralNet restoreNeuralNet(String fileName) {
try {
      FileConnection fconn =
```

```
                        (FileConnection)Connector.open(fileName);
        InputStream in = fconn.openInputStream();
        byte[] netData = new byte[(int)fconn.fileSize()];
        in.read(netData);

        NeuralNet net = new NeuralNet();
        net.resurrect(netData);
        return net;
} catch (Exception ex) {
        ex.printStackTrace();
        return null;}}
```

The same as with Serialization mechanism above code is generic and it does not depend on the internal structure of the saved/restored neural network. Such restored network is ready to be used. It is the exact copy of the original network built with jMENN Neural Network Editor with its internal objects being modified to be compatible with CLDC 1.1 and MIDP 2.0 profiles. ready to be used.

6.4.2   Using the neural network

In general, when embedding the neural network into an application, an embedding application have to be able to supply data to a neural network and read its results. For instance, it must be able to write data to a buffer, interrogate the network, and read its results at the end of data elaboration. With jMENN there are two ways to accomplish it. Depending on how the embedding application will interrogate the network, it can choose between synchronous and asynchronous mode.

6.4.2.1   Synchronous Mode

In synchronous mode, the network is interrogated with multiple input patterns, all collected and stored in memory before network interrogation. To embed the network in synchronous mode a `MemoryInputSynapse` and `MemoryOutputSynapse` objects must be used. They hold a reference to data input and output array buffers. The J2ME code that embeds the network into an application is illustrated in the following listing.

```
private double[][] inputArray = ((0, 0),(0, 1),(1, 0),(1, 1));

private MemoryInputSynapse memInp;
private MemoryOutputSynapse memOut

private void EmbedSynchronous(NeuralNet nnet) {
if (nnet != null) {
        Layer input = nnet.getInputLayer();
        input.removeAllInputs();

        memInp = new MemoryInputSynapse();
        memInp.setFirstRow(1);
        memInp.setAdvancedColumnSelector("1,2");
        input.addInputSynapse(memInp);
```

```
        Layer output = nnet.getOutputLayer();
        output.removeAllOutputs();

        memOut = new MemoryOutputSynapse();
        output.addOutputSynapse(memOut);

        nnet.getMonitor().setTotCicles(1);
        nnet.getMonitor().setTrainingPatterns(4);
        nnet.getMonitor().setLearning(false);
    }
}
```

First of all, all the existing I/O components linked with the input and output layers of a recently resurrected network must be removed. It ensures that none of the I/O components used in the editor to train the network are unintentionally left. In their places a `MemoryInputSynapse` and `MemoryOutpuSynapse` are connected and configured to run mode. In order to interrogate a simple elaboration of data, the embedding application have to call the method similar to this:

```
private void elaborate() {
        memInp.setInputArray(inputArray);
        nnet.go();
        for (int i=0; i < 4; ++i) {
                double[] pattern = memOut.getNextPattern();
        }
        nnet.stop();
}
```

To provide the neural network with the input patterns, the embedding application must call the `MemoryInputSynapse.setInputArray` method, passing a predefined 2-dimensional array buffer `inputArray`. In turn, in order to get the resulting patterns from the recall phase it calls the `MemoryOutputSynapse.getNextPattern` method. This method waits for the next output pattern from the net and returns an array of doubles containing the response of the neural network. This call is made for each input pattern provided to the net.

6.4.2.2   Asynchronous Mode

In asynchronous mode, an external asynchronous source of data interrogates the net with an input pattern. To embed the network in asynchronous mode a `DirectSynapse` object must be used as input and output instead. The following code illustrates how to embed a network in asynchronous mode:

```
private void EmbedAsynchronous(NeuralNet nnet) {
        if (nnet != null) {
                Layer input = nnet.getInputLayer();
                input.removeAllInputs();

                DirectSynapse memInp = new DirectSynapse();
                input.addInputSynapse(memInp);

                Layer output = nnet.getOutputLayer();
```

```
            output.removeAllOutputs();

            DirectSynapse memOut = new DirectSynapse();
            output.addOutputSynapse(memOut);
        }
    }
```

The `DirectSynapse` object is not an I/O component, as it doesn't inherit the `StreamInputSynapse` class. Consequently, it doesn't call the `Monitor.nextStep` method, so the neural network is no more controlled by the Monitor's parameters. It is now the embedding application role to control the neural network. It must know when to start and stop it, etc. For the same reasons, the 'TotCycles' and 'Patterns' parameters of the `Monitor` object don't need to be set.

To be able to interrogate the network in asynchronous mode first the call to the `NeuralNet.start` method must be made. To perform a simple elaboration the net takes one pattern for each query, using the `DirectSynapse.fwdPut` method, which as its parameters accepts a `Pattern` object. As a result, to retrieve the output pattern from the network the application call the `DirectSynapse.fwdGet` method as illustrated in the following code example:

```
 nnet.start();

 for (int i=0; i < 4; ++i) {
        Pattern iPattern = new Pattern(inputArray[i]);

        iPattern.setCount(1);

        memInp.fwdPut(iPattern);

        Pattern pattern = memOut.fwdGet();

        System.out.println("Output"+i+" ="+pattern.getArray()[0]);
 }
```

Asynchronous mode without a central control has some advantages over the synchronous mode. The neural network in asynchronous mode can be trained on remote machines. Therefore such system can either use the resulting patterns from an originally trained neural network to resolve stated problem, what ends neural network lifecycle, or by connecting a teacher synapse build an adaptation mechanism. Such mechanism, following some predefined adaptation protocols, will modify internal weights structure of a neural network.

6.5   Mobile Edition source code modifications

Java technologies has been grouped into three editions: Micro Edition (J2ME technology), Standard Edition (J2SE technology), and the Enterprise Edition (J2EE technology). The smallest in its footprint J2ME focus on applications for small/wireless devices, such as

cellular phones, pagers, smart cards, and personal assistant devices (PDAs). The J2SE platform centres on Java applications for desktop computers, servers, as well as powerful embedded environments. The Java EE, in turn, is the industry standard for enterprise Java computing to create next-generation web applications, and enterprise solutions.

Figure 6.5 Java 2 Platform overview includes Micro Edition (J2ME technology), Standard Edition (J2SE technology), and Enterprise Edition (J2EE technology) (Oracle Corporation 2010).

Because "one size doesn't fit all" applications nor devices, these platforms differs in terms of libraries and tools that they provide, in order to best answer application's and user's requirements. Focusing on the J2ME, its profiles specifies the minimum sets of APIs for particular devices. It aims to optimize the Java virtual machines and programs to become very small and able to embed inside a small, tiny devices. On one hand this approach brings advantages in terms application performance, resources utilisation or programs size but on the other hand it also imposes some limitations on the application development process and cross platform compatibility issues. This implications are at the root of JOONE source code modifications and new jMENN release for J2ME platform discussed in this section.

6.5.1 CLDC vs. J2SE Java Virtual Machine

A number of features commonly known from J2SE platform have been removed in J2ME virtual machine supporting CLDC. This is either because they were too expensive to implement or their presence would impose security problems. Therefore the virtual machines supporting CLDC platform has the following limitations, where *) highlights those that are relevant to jMENN port:

- **No support for reflection*):** Because reflection is not supported, the `Serializable` interface is not supported.

- **No finalization:** The CLDC APIs do not include the `Object.finalize()` method so the final clean-up operations cannot be performed on object data before the object is garbage collected.

- **Limited error handling:** Runtime errors are handled in an implementation-specific manner. The CLDC defines only three error classes: `java.lang.Error`, `java.lang.OutOfMemoryError`, and `java.lang.VirtualMachineError`. Non-runtime errors are handled in a device-dependent manner that involves terminating the application or resetting the device.

- **No Java Native Interface (JNI):** A Java virtual machine supporting the CLDC does not implement the Java Native Interface (JNI) primarily for security reasons. Also, implementing JNI is considered expensive, given the memory constraints of CLDC target devices.

- **No user-defined class loaders:** A Java virtual machine supporting the CLDC must have a built-in class loader that cannot be overridden or replaced by the user. This is mainly for security reasons.

- **No thread groups or daemon threads:** While a Java virtual machine supporting the CLDC implements multithreading, it cannot support thread groups or daemon threads. If thread operations for a group of threads must be performed, the application must use a collection objects to store the threads.

- **Limited number of system, I/O, and utility classes from the J2SE platform*):** These classes are located in the `java.lang`, `java.io`, and `java.util` packages and are derived from JDK1.3 APIs.

- **I/O and utility classes commonly known from the J2SE platform that are part of CLDC and MIDP specific packages*):** These classes are located in the `javax.microedition` package and its sub-packages.

- **No Property Support*):** In the CLDC, there is no implementation for the `java.util.Properties` class, however, the set of system properties is available. These can be accessed by calling the `System.getProperty(String key)` method.

Above limitations refer to the CLDC 1.1 (JSR 139 specification) which is a revised version of the CLDC 1.0 specification, and includes new features such as floating point and weak reference support, in additional to other enhancements. CLDC 1.1 is backward compatible with CLDC 1.0, and continues to target small and resource-constrained devices with the objective of maintaining a tight footprint. Limitations that are relevant to the jMENN port are discussed in details below.

6.5.2   No reflection support

Reflection mechanism provides automatic objects serialization. Any object in Java SE that needs to be serialized to a byte stream and then stored to a file, or any other persistent data format must implement `Serializable` interface. Class implementing the interface marks itself as "okay to serialize," and Java then handles serialization internally. There are no serialization methods defined on the `Serializable` interface, but classes that require special handling during the serialization and deserialization process must implement `writeObject` and `readObject` methods. The `writeObject` method is responsible for writing the state of the object for its particular class so that the corresponding `readObject` method can restore it. The `readObject` method is responsible for reading from the stream and restoring class fields.

Unlike Java Standard Edition the Connected Limited Device Configuration (CLDC) does not support reflection mechanism; therefore automatic serialization is not possible. CLDC application that needs to persist objects must build its own persistence mechanisms. It is difficult to build general persistence support that works with arbitrary objects. However, writing task-specific persistence mechanisms is much simpler and arguably more suited to the constraints of the CLDC platform.

Such persistence mechanism is easier to implement with the cooperation of the class that is to be persisted. In order to enable objects persistence to byte stream the following `Persistent` interface was defined:

```
import java.io.*;

/**
 * A simple interface for building persistent objects on platforms
 * where serialization is not available.
 */
public interface Persistent {

/**
* Called to persist an object.
* @return serialized object as byte[] array.
* @throws IOException
*/
byte[] persist() throws IOException;

/**
* Called to resurrect a persistent object.
* @param data serialized object as byte[] array.
* @throws Exception
*/
void resurrect(byte[] data) throws IOException, Exception;
}
```

`Persistent` interface provides two methods: `persist` and `resurrect`. The first one, like `writeObject` methods from `Serializable` interface, recursively saves all needed attributes of the class, while the second one, like `readObject` method from `Serializable`

interface, recursively loads them from the byte stream. The difference is that this process is no longer automatic and each class has to have a special handling during the persistence process. Moreover each class that needs to be persisted must have a null constructor. Let's consider a simple class with number of different type attributes that implements a `Persistent` interface:

```java
import java.io.*;

public class SampleClass implements Persistent {

private boolean      sampleBoolean;
private int          sampleInt;
private double       sampleDouble;
private String       sampleString;
private CustomObject sampleObject;
private int[]        sampleArray;
private Vector       sampleVector;

// MUST have a null constructor
public SampleClass(){
}

// class methods here
// ...

public byte[] persist() throws IOException {
        ByteArrayOutputStream bout = new ByteArrayOutputStream();
        PersistStream dout = new PersistStream(bout);

        dout.writeBoolean(sampleBoolean);
        dout.writeInt(sampleInt);
        dout.writeDouble(sampleDouble);
        dout.writeUTF(sampleString);
        dout.writeObject(sampleObject);
        dout.writeArray(sampleArray);
        dout.writeVector(sampleArray);

        dout.flush();
        return bout.toByteArray();
}

public void resurrect(byte[] data) throws IOException{
        ByteArrayInputStream bin = new ByteArrayInputStream(data);
        ResurrectStream din = new ResurrectStream(bin);

        sampleBoolean = din.readBoolean();
        sampleInt = din.readInt()
        sampleDouble =din.readDouble();
        sampleString = din.readUTF();
        sampleObject = (CustomObject)din.readObject();
        sampleArray = din.readIntArray1D();
        sampleVector = din.readVector();

        din.close();
        bin.close();}
}
```

Persistence is accomplished using the `PersistStream` and `ResurrectStream` objects, which extend `DataInputStream` and `DataOutputStream` respectively. This stream objects allow easily writing and reading Java primitives, strings, vectors, arrays and objects, assuming these objects implements `Persistent` interface too. Each class attribute has to be manually written out or read from a byte stream. Notice that the order in which the attributes are persisted and resurrected does matter and cannot be changed. Any change to the order will cause and `Exception` to be thrown by the `resurrect` method.

With the SampleClass, an sample object can be persisted at any point, like this:

```
SampleClass sample = .....; // SampleClass instance

try {
        byte[] persisted = sample.persist();
} catch( java.io.IOException e ){
        // do something here
}
```

Restoring the SampleClass is done like this:

```
byte[] persisted = ....; // persistence info
SampleClass sample = new SampleClass();

try {
        sample.resurrect( persisted );
} catch( java.io.IOException e ){
        // do something here
}
```

Things are more complicated, of course, if the objects contain references to other persistent objects. The example above includes a `CustomObject` which use a reference to another persistent object. This complicates the persistence because, instead of persisting a single object, it needs to persist a sequence of objects. This can possibly contain a cross-references to each other, what complicates things even more.

`PersistStream` and `ResurrectStream` preserve cross-reference. The streams traverse the object graph, ensure that each object gets written at most once, and deal with cycles in the graph. Each stream maintains its internal register of objects that were already persisted or resurrected. The register is the `Hashtable` where objects and theirs unique identifiers in byte stream are stored. Each object gets written to the byte stream and a register only when it first appears in the stream. This is accompanied by the unique identifier generation that is written to the register as well as to the byte stream just before the object data. Each consecutive occurrence of the same object instance in the stream results in substitution of the object by its unique identifier that is taken from the register and written to the byte stream. The same but reversed cross-reference check and register are used to restore the objects out of the byte stream.

6.5.3 Limited set of system, I/O, and utility classes

The J2SE APIs require several megabytes of memory, and therefore, they are not suitable for small devices with limited resources. When designing the APIs for the CLDC, the aim was to provide a minimum set of libraries useful for application development and profile definition for a variety of small devices. The extension layer for standard libraries is a profile which is a set of vertical APIs that reside on top of a configuration to provide domain-specific capabilities. In order to provide a flexibility and support for some common problems the J2ME platform provides also some optional APIs, however this are very often device and profile dependent. Therefore, CLDC application developers, who need certain libraries or even single classes, which are not supported on the platform they are targeting, very often must build their own solutions, or use third party library, such as J2ME Polish (Enough Software 2004). It is a suite of tools and technologies that will "polish" the standard J2ME libraries adding necessary functionality. In this project a custom package called j2megloss has been designed that is used to replace missing functionality.

The CLDC library APIs can be divided into the following two categories:

- Classes that are a subset of the J2SE APIs: The CLDC inherits a number of system, I/O, and utility classes from the J2SE platform. These classes are located in the `java.lang`, `java.io`, and `java.util` packages and are derived from JDK1.3 APIs.

- Classes specific to the CLDC and MIDP: These classes are located in the `javax.microedition` package and its sub packages.

Please note, that each class that has the same name and package name as a J2SE class must be identical to, or a subset of, the corresponding J2SE class. The semantics of the classes and methods cannot be changed, and the classes cannot add any public or protected methods or fields that are not available in the corresponding J2SE class libraries.

The following table provides a detailed list of all compatibility issues with regards to jMENN engine port. It specifies the J2SE interfaces and classes that are not supported on CLDC platform, but which are present in the original JOONE engine. During the porting process this classes must have been either replaced or ported or supplement by J2ME compatible classes. Some of them are part of CLDC and MIDP specific packages and their substitution was fairly easy. Things got complicated when the missing classes were not part of any CLDC capable library. These classes must have been either implemented from scratch or ported to the CLDC platform. They are most often part of the `org.j2megloss package`. Note that many of the classes are ports of J2SE corresponding classes. Since Java is an open source platform, many of them are the optimized versions of their original implementations. There is also one external CLDC library, namely jazzlib (Leuner 2004) that provides an implementation of the `java.util.zip` classes. This is used to encode/decode the PNG file format. The

details of all substitutions in the jMENN engine source code in comparison to JOONE engine source can be found in the Table 1,2 and 3 in Appendix 3.

6.5.4 No Properties support

In the CLDC platform, there is no implementation for the `java.util.Properties` class. However, the set of system properties shown in Table 6.2 is available. They can be accessed by calling the `System.getProperty(String key)` method.

Key	Description	Default Value
`microedition.platform`	The host platform or device	`null`
`microedition.encoding`	The default character encoding	`ISO8859_1`
`microedition.configurations`	J2ME configuration and version	`CLDC-1.1`
`croedition.profiles`	Name of supported profiles	`null`

Table 6.1 Standard system properties for CLDC platform

When an application needs to store some properties a custom `org.j2megloss.Properties` class can be used. It provides the interface to save/load application specific properties to/from the record store of a unique name. The `org.j2megloss.Properties` class provides a support for the jMENN Neural Network properties. The `Properties` interface presents as follow:

```
public class Properties{
  Properties(String recordStoreName);
  public byte[] get(String key) ;
  public void put(String key, byte[] value);
  public int numberOfRecords() ;
  public void clear();
  public Object[][] getAll() ;
  public boolean remove(String key);
  public boolean contains(byte[] value;
  public String getKey(byte[] value);
}
```

A record store consists of a collection of records, which will remain persistent across multiple invocations of the MIDlet. A record store allows storing multiple records, each as an array of bytes. There might be multiple record stores, each containing number of records as long as they are given different unique names.

The fields of `Properties` class are persistent as long as the MIDlet suite is not removed. This is a responsibility of a platform to make its best effort to maintain the integrity of the record stores throughout the normal use of the platform, including reboots, battery changes,

etc. When the MIDlet suite is removed from a platform all the record stores associated with it will also be removed.

6.5.5   JOONE classes not supported by jMENN

During the optimisation process a number of classes have been selected as to expensive or not necessary for the joone Mobile Edition Neural Network Editor. These classes were removed from the jMENN core engine library. However for the backward compatibility with JOONE 2.0.1 RC1 version the removed classes are part of the jMENN Neural Network Editor. The following classes, type of classes or packages were removed from the jMENN core engine:

- `BeanInfo` classes: it described how the network component appears in a Neural Network Editor tool. A Neural Network Editor can query the `BeanInfo` to find out which properties it should display, what methods the component provides and what events it triggers.

- *`org.joone.io`*
  - `XLSInputSynapse`: This class allows data to be presented to the network from an Excel XLS formatted file. The XLS file name must be specified and a worksheet name is optional.
  - `XLSInputTokenizer`: This class is an input tokenizer for XLS file format.
  - `XLSOutputSynapse`: This class allows data to be written to the Excel XLS formatted file.
  - `YahooFinanceInputSynapse`: The YahooFinanceInputSynapse provides support for financial data input from financial markets. The synapse contacts YahooFinance services and downloads historical data for the chosen symbol and date range.

- *`org.joone.log`*
  - `Log4JLogger`: Logger that uses Apache's Log4J to log the messages.

- *`org.joone.script`*
  - `JooneGroovyScript`: Scripting class based on the groovy scripting mechanism.
  - `JooneScript`: JOONE scripting mechanism.

- *`org.joone.util`*
  - `NeuralNetRunner`: application to run a neural network as a service.
  - `GroovyMacroPlugin`: Macro plug-in base on the groovy scripting mechanism.

- *`org.joone.helpers.templating`:* This Class generates the source code to build the neural network passed as a parameter to the `getCode()` method, according to a predefined template. It searches the file containing the template either in the file system or in the class path.

6.6   jMENN Converter plug-in

Recently released Neural Network Editor jMENN version provides a built in export and import plug-in which enables to export the neural network to three main data types:

- *.snet* (serialized file format) which is Java Standard Edition compatible format. It enables to serialize/deserialized a neural net objects, which can be then further embedded into an existing or new application or system running on Java SE platform.

- *.pnet* (persisted) file format which is Java Mobile Edition compatible format. It enables to persist/resurrect a mobile version of a neural network. This can be further embedded into an existing or new mobile application running on Java ME CLDC platform.

- *.xml* (extensible mark-up language) file format which is an open system compatible format. It enables to save/load a neural net objects and attributes to a xml file. This can be used to embed the network into an existing or new application running on Java EE platform such as a web servers, databases or easily send it over the network.

The export process is very simple. In order to export the network to any of these files a user has to select: **File -> Export Neural Net as …** and choose the file format of interest as illustrated in figure 6.6a.

In turn, to import the network from any of these files a user has to select: **File -> Import Neural Net from…** and using a file browser choose the file of interest that should be imported. The file must be in any of the supported file formats The process is illustrated in figure 6.6b.
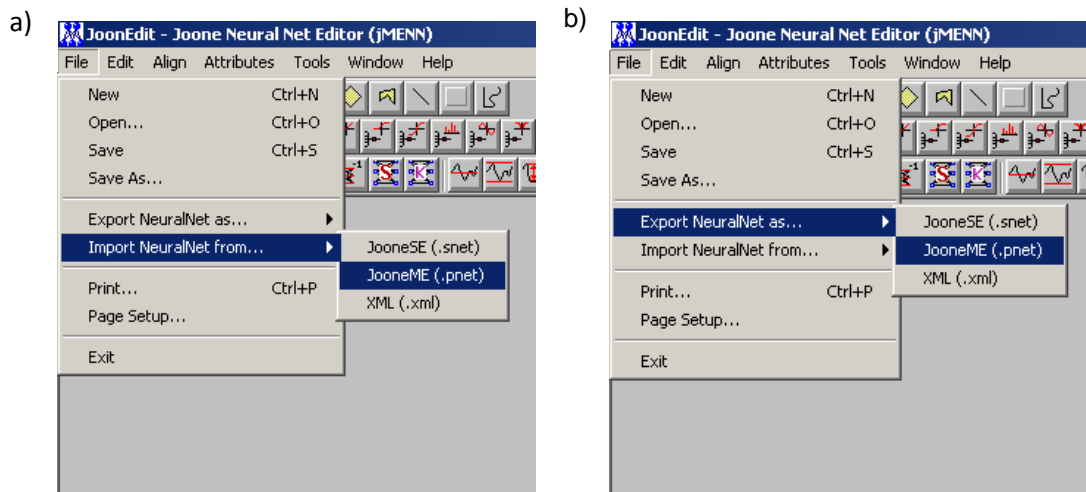
Figure 6.6 Neural Network Editor (jMENN version) process to a) Export Neural Net as …; b) Import Neural Net from …

Chapter

7

Vital Signs Classification and Decision Process

7.1 Introduction

This chapter focuses on data processing model and its building blocks. The methods proposed here are based on mixture of statistical and machine learning algorithms and were developed using the jMENN and life-cycle model presented in chapter 6. The chapter describes how concurrent analysis of multiple physiological signals has been accomplish in order to achieve adaptive, patient-specific well-being monitoring model. Associated implementations, including testing, training and validation of the model, are presented.

7.2 Ubiquities well-being analysis concept

Smart Wearable Systems (SWS) in order to serve as an autonomous Personalised Health Assistants (PHA) require a classification, prognosis and decision model. Such models are crucial in early identification of medical conditions and help to facilitate conventional clinical diagnosis processes, by analysing environmental and physiological data and providing intelligent diagnostic assessment and alert feedback, either to the patient or directly to the healthcare professionals. In the long term, it is envisaged that wide spread adoption of such automated personal health assistants, capable to integrate multiple sensing devices and parameters as well as autonomously make intelligent decisions, could contribute to a substantial reduction of health care costs as well as on the time for delivery of care, translating consequently, in an improvement in quality of life and well-being of the population.

The problem current PHA's is their robustness and accuracy at the affordable cost. This is currently difficult to achieve when the monitoring is performed during normal daily activities with sensing devices embedded into daily necessity items such as watches or clothing. The requirement for such ubiquitous models it to constantly deliver trustworthy analysis leading to personalised health feedback even in presence of unexpected artefacts such as body movements, background environmental noise or accidental errors. All these issues increase the complexity and cost of analysis, especially when such analysis should be performed in a timely manner. On the other hand, portable health monitoring devices are strictly restricted in size,

weight and power consumption. These, in turn, require systems and models to be flexible and adaptable but also compact and inexpensive at the same time.

A trade-off between robustness, complexity of processing, mobility and cost can be achieved with the smartphone embedded Inference Engine model, what this chapter attempts to prove. Presented in previous chapter, the Inference Engine platform combines recent advances in fundamental capabilities of the handheld system with the accuracy of the state-of-the-art inference algorithms and enables rapid model development, remote deployment and future reconfiguration at the affordable cost and effort. It allows to shift from passive patient monitoring devices with fixed functionality to a proactive agents that may interact with the patient and autonomously reconfigure to provide the most accurate information on patient health status.

In order to fully exploit the advances of the proposed framework, a preventive well-being analyses model have been developed to perform ubiquities, multi-parametric and adaptive patient-specific monitoring that can answer shortcomings of traditional healthcare models. The model presented in this chapter combine single physiological signals analysis with multi parameter signal analysis using both supervised and unsupervised learning algorithms. As a result a new patient-specific two-stage distributed event-triggered hybrid model with context adaptation is proposed. It allows to detect any abnormalities by continuously monitoring and analysing five primary vital signs, recognised as standards in most medical settings. These are: Body Temperature (TEMP), Electrocardiogram (ECG) with Heart Rate (HR), Blood Pressure (BP), Respiratory rate (RESP), and Pulse oximetry (SpO2). Data fusion is achieved through Self-Organizing Maps (SOM) with descriptive statistics such as measures of central tendency and spread for adaptation purposes. The traditional threshold based models, deployed as set of distributed sensor-based event dispatchers, are used as the reference point for user adaptation. The whole model as well as its individual components were trained and tested on representative examples of ECG signals from MIT-BIH Arrhythmia database (Moody and Mark 2001) as well as multi-parameter data from MIMIC database (Moody and Mark 1996) obtained from Physiobank (Goldberger, Amaral, Glass *et al.* 2000). The model capabilities to detect health status deterioration was evaluated against results from existing threshold based multi-parameter patient ICU monitors, as a scenario reach in clinically significant health events. The proposed model and methods compares favourably with published results for other multi-parameter signal analysis, and clearly proves superior to threshold based monitors. The experimental results show high accuracy of the proposed approach in identifying potential health risks due to health status deterioration. The model proves also to be effective in reducing the false alarms caused by patient movements, monitor noise, or imperfections of the standard population based diagnostic schemes.

7.3   Model overview

Preventive measures in the traditional healthcare model are primarily based on periodically scheduled evaluations during clinic visits that are intended to detect the onset of an illness. Such visits often present an incomplete assessment of the patient's health by providing only instantaneous picture of the patient's state, since signs that might present themselves frequently during normal daily activities may disappear while the patient is hospitalized, causing unnecessary costs and diagnostic difficulties. On the other hand some symptoms of a medical condition may only be acquired in a natural environment, such as home or in the workplace (Shephard 2003). As a result missed or late detected symptoms affect the early identification of illness and delay prevention of any unsought trends changes. All these degrade the outcomes of treatment and eventually increases long-term healthcare costs (Polisena, Tran, Cimon *et al.* 2010, Martin-Lesende, Orruno, Bilbao *et al.* 2013).

Most of the current wearable health monitoring devices, indeed, are capable to provide continuous real-time analysis of one or, less frequently, many vital signs with high accuracy, though their current assessment mainly relies on comparing the measured values with medically approved population based normal ranges or classification rules. Firstly because of the complexity of the human body, the monitoring of individual's health status by focusing on a single dominant physiologic signals in separation is not satisfactory. Secondly, because every human being is different, performing a simple rule-based classification by referring to population based diagnostic methods only, cannot lead to an accurate individualised medical diagnosis of subtle conditions. Therefore accepted ranges for each physiological signal are recognised as a valid coarse approximations only and ranges considered as good will vary significantly with every human. They do however set an orientation point for further analysis and modern more sophisticated signal processing models, in order to be valid, should take these ranges into account. They can server, for example, as a good starting point for adaptation to an individual's physiological characteristics or different diagnostic needs.

Proposed here patient adaptation can enhance the diagnostic power of such models by correlating physiological data with contextual information such as patient's activity, environment or social interaction. This enable to model the individual's personal physiological characteristic and observe how it evolves over time. In this approach, an physiological signature of the individual, called Health Index, is compiled by aggregation of different features from the collected signals and is used in the monitoring phase as a reference point for detection of medical abnormalities.

Proposed hybrid approach, illustrated in Figure 7.1, integrates simple rule-based classification of individual sensor data with adaptive analysis of physiological signature of the individual. As in standard data processing model, the monitoring flow starts with data aggregation element, which samples the amplitudes of the analogue signals at regular, uniform intervals, and quantize each sample to the nearest value within a range of digital steps. Such
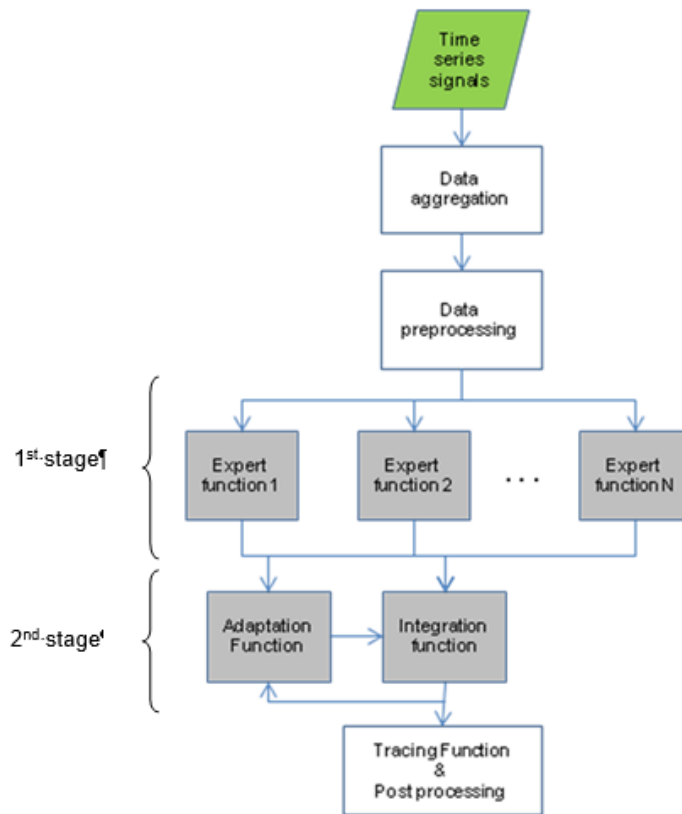
Figure 7.1 Data flow chart for patient-specific vital signs acquisition and processing.

signal is then feed to the initial pre-processing element. The pre-processing element normalise and converts raw date to time series with appropriate units.

The first stage of the signal analysis is done by the Expert Function (EF) elements, which are signal specific components of the proposed model, responsible for feature extraction tasks and their classification against approved ranges. The EF elements are distributed software blocks which operates partially on sensor nodes and partially on personal server device. Such distribution of processing tasks is dictated by the computational efficiency and capabilities of each device. Each expert function is designed as a coarse-grained configurable data flow element, for which internal functionality can be altered/extended when necessary, without affecting the entire system operation.

The expert function elements support (i) sentinel events generation, based on common threshold values from physiological signals scoring systems used to identify patients at risks and (ii) signal-specific algorithms including new QRS detection algorithm and binary heart beat classification algorithm. Moreover each EF element generates notifications and alarms in case of any anomaly or abnormality detection. This is based on the NEWS scoring system which is discussed in the section 7.4. The NEWS schema is also used on the senor nodes to decide whether a samples should be sent, for instance to Personal Server or further to medical professionals for further analysis, or simple local calculation of correlation and/or dispersion coefficients on the sensor node is satisfactory for keeping the integrative model up to date.

The outputs from multiple Expert Functions link with the Integration Function (IF) element for concurrent, integrative analysis of all physiological data streams obtained from wearable sensors. By combining multiple vital signs together, we are able to construct more accurate, unified representation of an individual's health status. This is achieved by reducing the dimensionality of the monitored data using Self-Organizing Map (SOM) algorithm, which belongs to the class of unsupervised and competitive learning algorithms. It is a "sheet-like" neural network, with nodes arranged as a regular, usually two dimensional grid, where each node is directly associated with a weight vector of the same dimension as the input vector.

The choice of Self-Organizing Maps is motivated by its unsupervised ability to adjust these weight vectors until the map represents "a picture" of the input data set. Unlike other learning technique in neural networks, training a SOM requires no target vector, which in our application would be very difficult to obtain. Instead, where the node weights match the input vector, the area of the lattice is selectively optimized to more closely resemble the data for the class that the input vector is a member of. The neuron whose weight vector is most similar to the input is called the best matching unit (BMU). From an initial distribution of random weights, and over many iterations, the SOM eventually settles into a map of stable zones. The zones are effective feature classifiers. Any new, previously unseen input vectors presented to the network will stimulate nodes in the zone with similar weight. The objective is to achieve such configuration in which the distribution of the data is reflected and the most important metric relationships are preserved. In particular, we are interested in obtaining a correlation between the similarity of items in the dataset and the distance of their most alike representatives on the map. In other words, items that are similar in the input space should map to nearby nodes on the grid, building in such a way the health status map of an individual. In such model any abnormal conditions or emergency situations will result in deviations with respect to normal conditions, associated with a certain area on the map.

The SOM based Integration Function, enhances the traditional quantitative inspection of vital signs, enabling to focus on the evolution of the map, and therefore the evolution of the human health status. Depending on the nature of change and its time span, this can be characterized in two ways.

Firstly, abrupt and short term changes outside the generally accepted zones may be an indicator of abnormal conditions and are controlled by a trajectory of a BMU on a two-dimensional Self-Organizing Map. An example could be the input sensor dataset changes over time due to illness. In this case the best matching unit (BMU) for the items in a dataset will move to a new location in the map. The role of the Tracing Function (TF) is the presented model is to track the locations of the BMU and visualise these movements by showing a trace of the health status that shows what path it has taken. This does not only allow to inspect what was the cause(s) of improvement or deterioration but also to determine correlations between certain physiological parameters what when deeply analysed could improve our understanding of the cause and effect relationships amongst vital signs (or other parameters). From the diagnostic

point of view it could be especially interesting to overlay multiple subjects with similar chronic conditions in order to view groups of traces and to analyse their relationships.

Secondly, gradual and long term changes within generally accepted ranges may be considered as the natural evolution of the health status for instance due to aging. It may also be a positive trend when for example due to long lasting, regular physical activity the health status improves. In this case the health status map should evolve as well. The role of the Adaptation Function (AF) in the presented model is to control the adaptation of Integration Function (IF) model to the characteristic of a specific patient. Unlike the initial standard, population based training of the SOM algorithm with time decreasing learning parameters, the proposed adaptation function (AF), based on the TOSOM algorithm proposed by Shah-Hosseini and Safabakhsh (Shah-Hosseini and Safabakhsh 2000), use the time-independent learning parameters, which changes their values according to the conditions of the incoming samples, and not to the elapse of time. For this purpose, each neuron, independently from other neurons in the map, has its own learning rate and neighbourhood size, which control the adaptation.
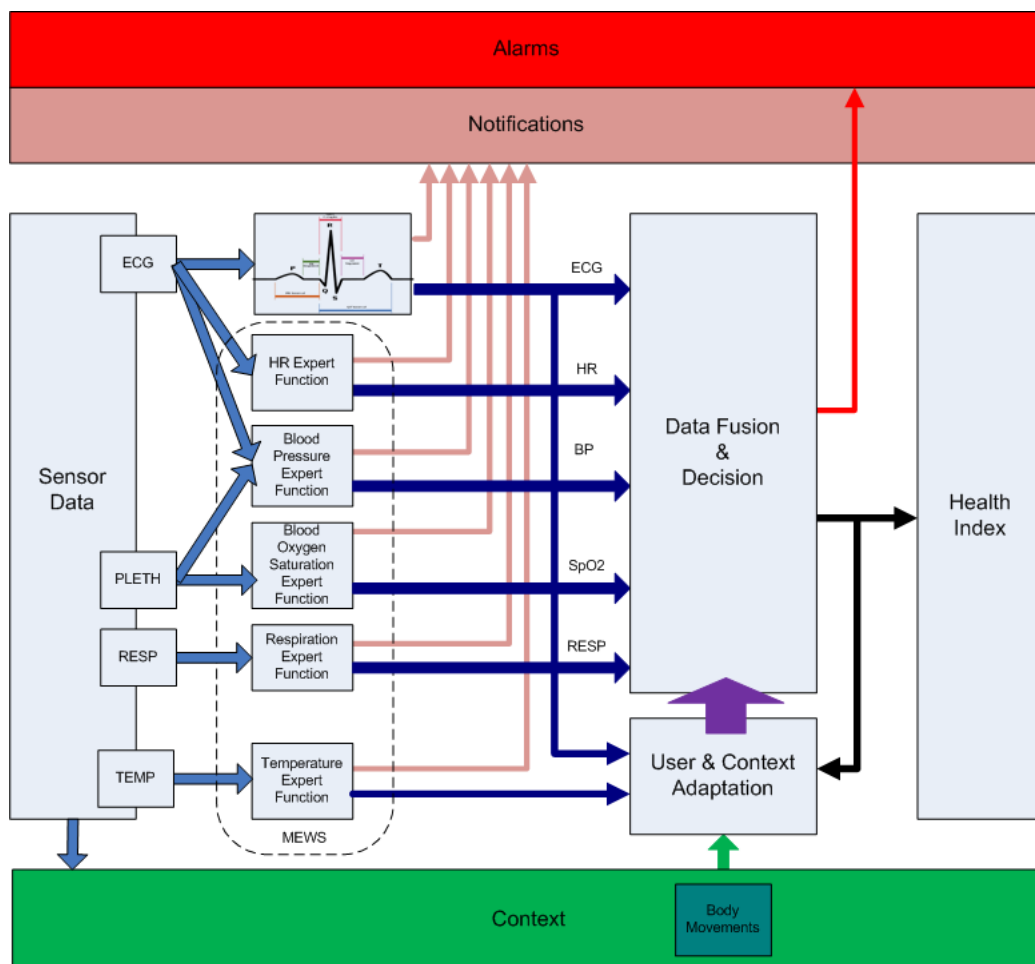
Figure 7.2 An architecture of the proposed hybrid model for patient-specific well-being analysis.

The learning rate of each neuron is considered to follow the values of a function of distance between the input vector and its synaptic weight vector. In turn, the neighbourhood size of each neuron is considered to follow the distance between the neuron's synaptic weight vector and the weight vectors of its neighbouring neurons.

Proposed model allows us to treat the SOM-based data analysis process as an interactive and iterative process of repeatedly training, viewing and analysing the map which can be considered as a physiological signature of the patient being monitored which aim to improve the robustness and accuracy of the assessment model. Figure 7.2 illustrates the complete architecture of the proposed patient-specific well-being analysis model. The remainder of this chapter discuss each of these elements in details.

7.4   Threshold based analysis

Health status can be quantified by measurement of a combination of simple physiological parameters such as respiration rate, oxygen saturations, temperature, systolic blood pressure, pulse rate and level of consciousness – all of which are easily recorded during routine patient assessment (Royal College of Physicians 2012). Based on these simple physiological measurements, there are now many 'early warning scores' or 'track-and-trigger systems' in use worldwide (Jansen and Cuthbertson 2010). These vary in complexity ranging from single physiological parameter scores through to multiple parameter, aggregate-weighted and a combination of these systems (Smith, Prytherch, Schmidt *et al.* 2008a, Smith, Prytherch, Schmidt *et al.* 2008b, Patterson, Maclean, Bell *et al.* 2011). These scores have been used to determine the speed and level of clinical response required for an individual patient. In the setting of acute illness, these scoring systems have also been shown to be a good predictor of patient mortality and hospital length of stay (Subbe, Kruger, Rutherford *et al.* 2001, Morgan and Wright 2007, Gao, Mcdonnell, Harrison *et al.* 2007).

7.4.1   Modified Early Warning Score (MEWS/EWS)

One of the earliest and most common threshold based physiological signals scoring system to identify patients at risks is the Early Warning Score (EWS) (Morgan and Wright 2007). It is a score, track and trigger system that is calculated by nursing staff from the observations taken, to indicate early signs of a patient's deterioration. It is an additional tool that is utilised in conjunction with clinical judgement about the patient's condition, to facilitate detection of a deteriorating patients.

Morgan et al. (Morgan, Williams and Wright 1997) were the first to describe an EWS system, which has been broadly adopted in its modified early warning score version (MEWS; Table 7.1). It is based on data derived from four physiological readings: systolic blood pressure, heart rate, respiratory rate and temperature as well as observation of level of consciousness

using AVPU scale (an acronym from "alert, voice, pain, unresponsive"). It is a simple scoring system with an upper and lower score of 0–3 points attributed to each of these parameters from which a total score is calculated. In general, a score of five or more is statistically linked to increased likelihood of death or admission to an intensive care unit (Subbe, Kruger, Rutherford *et al.* 2001).

	3	2	1	0	1	2	3
Systolic BP (mmHg)	< 70	71-80	81-100	101-199	-	≥ 200	-
Heart Rate (bpm)	-	< 40	41-50	51-100	101-110	111-129	≥ 130
Respiratory Rate (bpm)	-	< 9	-	9-14	15-20	21-29	≥ 30
Temperature (ºC)	-	< 35	-	35-38.4	-	≥ 38.5	-
AVPU score				**A**lert	Reacting to **V**oice	Reacting to **P**ain	**U**nresponsive

Table 7.1 Modified Early Warning Scores (MEWS) in hospital admission (Subbe, Kruger, Rutherford *et al.* 2001).

In general, there is an identified threshold per parameter and/or the total score which, when reached, activates an escalation protocol. An escalation protocol, such as those in table 7.2-5, sets out the response required in dealing with different levels of abnormal physiological measurements and observations. This response may include:

- appropriate modifications to nursing care,
- increased monitoring,
- review by the primary medical practitioner, team or "on call team"
- calling for emergency assistance from intensive care or other specialist teams
- or activate the Emergency Response System.

As far as accepted ranges for each parameters are fairly the same worldwide, scoring systems and escalation protocols differ even between hospital amenities of the same country. An example of such variation are Modified Early Warning Score at Heart of England NHS Foundation Trust (Higgins, Maries-Tillott, Quinton *et al.* 2008) and Leeds Teaching Hospitals Trust (Von Lilienfeld-Toal, Midgley, Lieberbach *et al.* 2007) shown below.

Score Categories	3	2	1	0	1	2	3
Heart Rate (bpm)				51-100	101-110	111-129	>130
Systolic BP (mmHg)	<70	71-80	81-100	101 - 199		>200	
Respiratory Rate (bpm)		<8		9-16	17-20	21-29	>30
Oxygen Saturation (%)	<84	85-89	90-93	>94			
Temperature (ºC)		<35	35.1 - 36	36.1 - 37.5	37.6 – 38.1	>38.2	
Urine	Nil	1-20	21-35	No Concerns			
AVPU			Confusion/ Agitation	Alert	Voice	Pain	Unresponsive

Table 7.2 Modified Early Warning Score (MEWS) used at Heart of England NHS Foundation Trust (Heart of England Nhs Foundation Trust 2011)

LOW	MEDIUM	HIGH
MEWS 1-3 Or Patient causing CLINICAL concern	MEWS 4 or 5 Or Patient causing CLINICAL concern	MEWS 6 or MORE Or Patient causing CLINICAL concern
► Consider increasing frequency of observations ► Inform nurse in charge	► Inform nurse in charge ► Contact Junior Doctor to review patient within 30 minutes. If unable to attend contact Senior Doctor. ► In addition you must inform Critical Care Outreach team ► Initiate treatment and document all MEWS scores and any actions taken ► Continue to monitor patient observations every 30 minutes until MEWS below 4 ► If no improvement in MEWS score, seek Senior medical review	► Inform nurse in charge ► IMMEDIATE CALL to Critical Care Outreach team ► IMMEDIATE CALL to Junior Doctor to review patient within 30 minutes ► Inform patient's SPR / Consultant and ensure senior review within 1 hour ► Initiate treatment and document all MEWS scores and any actions taken ► Continue to monitor patient observations every 30 minutes until MEWS below 4 ► Consider transfer to high level of care

Table 7.3 Escalation protocol used at Heart of England NHS Foundation Trust (Heart of England Nhs Foundation Trust 2011)

Score	3	2	1	0	1	2	3
Heart Rate	<30	<40	40 - 50	51 – 100	101-110	111-130	>130
Systolic BP	<70	71 – 80	81 - 100	101 - 179	180 - 199	200-220	>220
Respiratory Rate		<8	8-11	12 - 20	21 - 25	26-30	>30
Oxygen Saturations	<85%	85-89%	90-94%	>94%			
Respiratory Support	BIPAP/ CPAP	60% or re-breathe bag	Oxygen Therapy				
Urine Output in last 4 Hrs	<80 mls	80-119 mls	120-200 Dialysis	201-799 New admission	>800 mls		
AVPU			New Confusion	Alert and responsive	Responds to Voice	Responds to Pain	Unresponsive

Table 7.4 Adult Modified Early Warning Score (MEWS) used at Leeds Teaching Hospitals NHS Trust (Leeds Teaching Hospitals Nhs Trust 2010)

Table 7.5 Escalation protocol used at Leeds Teaching Hospitals NHS Trust (Leeds Teaching Hospitals Nhs Trust 2010)

This policies were designed to provide evidence-based guidance on best practice when undertaking, documenting and responding to MEWS, to minimise the risk of patient deterioration and ensure that appropriate response and patient review takes place. However there is lack of consensus on what constitutes the 'ideal' early warning score system. There are more than 100 different types of chart where patients' vital signs are recorded, sometimes several and approach can lead to lack of consistency in detection and response to times a day. Comparing different systems in clinical use shows variation in which parameters are scored and how those scores are assigned to differing levels of deterioration (Prytherch, Smith, Schmidt *et al.* 2010). This variation in methodology acute illness for all grades of healthcare professionals what can result in poor health care and higher morbidity. This has led to a call in several countries for the development of a national early warning score, that would allow a standardised approach to assessing and responding to deteriorating patients.

7.4.2   National Early Warning Score (NEWS)

The Royal College of Physicians (RCP) has reviewed numerous MEWS implementations and concluded with justification that the routine recording of six physiological parameter, presented in Figure 7.3, should form the basis for the UK's National Early Warning Score (NEWS). All the six physiological parameters are readily measured in patients, either in the pre-hospital or hospital setting and can be repeatedly measured to document trends and assess changes in illness severity.

Respiratory rate

An elevated (tahypnea) or too low (braypnea) respiratory rate is a powerful sign of acute illness conditions, generalised pain or distress, in all patients.

Oxygen saturations

Oxygen saturations are a powerful tool for the integrated assessment of pulmonary and cardiac function.

Temperature

Both hyper- and hypothermia are included in the NEWS system reflecting the fact that the extremes of temperature are sensitive markers of acute-illness severity and physiological disturbance.

Systolic blood pressure

An elevated (hypertension) or too low (hypotension) blood pressure is an important risk factors for cardiovascular disease and may indicate cardiac failure or cardiac rhythm disturbance, amongst other syndromes.

Pulse rate

The measurement of heart rate is an important indicator of a patient's clinical condition. Both too fast (tachycardia) and too slow (bradycardia) heart rate may be normal with physical conditioning but it may also be an indicative of deteriorating health conditions.

Level of consciousness

Level of consciousness is an important indicator of health status. The assessment use Alert Voice Pain Unresponsive (AVPU) scale to measure a patient's level of consciousness. It is done in sequence and only one outcome is recorded.

Figure 7.3 National Early Warning Score (NEWS) physiological parameters.

As it is uncommon for significant disturbance of a single physiological parameter to occur in isolation, disturbances in multiple parameters are more common and an aggregate of the magnitude of disturbance is a more robust measure of acute-illness severity (Smith, Prytherch, Schmidt *et al.* 2008a, Prytherch, Smith, Schmidt *et al.* 2010). Whilst this is true for a single assessment at baseline, even more can be gained by repeated measurements to define trends which can highlight deterioration or improvement in a patient's clinical condition. Significant disturbances in these six parameters are not necessarily unidirectional, thus upward and downward trends needed to be weighted and scored. As a result the new standardise scoring chart has been proposed. The accepted ranges (excluding supplemental oxygen) presented in table 7.6 constitutes the ultimate reference point for further analysis conducted in this project.

Having defined the scoring template for NEWS, the RCP defined also the trigger thresholds for single parameters and for the aggregate score. It is critical to the performance of NEWS in terms of its ability to discriminate different levels of illness severity and also the frequency of urgent clinical reviews that would be triggered. Based on formal evaluation of the performance of the NEWS it was decided that an aggregate score of 5–6 should trigger a medium-level clinical alert, i.e. an urgent clinical review; and a NEWS score of 7 or more should trigger a high-level clinical alert, i.e. an emergency clinical review. It has been recommended that an extreme score of 3 in any one physiological parameter, should also trigger a medium-level alert, as presented in table 7.7.

PHYSIOLOGICAL PARAMETERS	3	2	1	0	1	2	3
Respiration Rate	≤8		9 - 11	12 - 20		21 - 24	≥25
Oxygen Saturations	≤91	92 - 93	94 - 95	≥96			
Any Supplemental Oxygen		Yes		No			
Temperature	≤35.0		35.1 - 36.0	36.1 - 38.0	38.1 - 39.0	≥39.1	
Systolic BP	≤90	91 - 100	101 - 110	111 - 219			≥220
Heart Rate	≤40		41 - 50	51 - 90	91 - 110	111 - 130	≥131
Level of Consciousness				A			V, P, or U

Table 7.6 The National Early Warning Score (NEWS) scoring system.

NEW scores	Clinical risk
0	Low
Aggregate 1–4	Low
RED score* (Individual parameter scoring 3)	Medium
Aggregate 5–6	Medium
Aggregate 7 or more	High

Table 7.7 The NEWS trigger system aligned to the scale of clinical risk.

Initial evaluations of the NEWS scoring and weighting system done with the vital signs database at Portsmouth Hospitals NHS Trust (Prytherch, Smith, Schmidt *et al.* 2010), proved NEWS system to be as good at discriminating risk of acute mortality as the best of the existing EWS systems. In turn the initial results from evaluation of the trigger thresholds, show that NEWS is likely to be more sensitive than most currently used systems at prompting an alert and clinical response to illness deterioration. It has been recognised that the overall performance of NEWS or any other EWS system is not solely dependent on the scoring system but the chosen outcome plus the sensitivity of the trigger thresholds and crucially, the organisation of the response. It is however envisaged that its huge potential added value is the national standardisation in assessment and response.

7.4.3 BAN distributed sentinel events detection model

Based on the NEWS scoring and weighting system, a BAN distributed sentinel events detection model is proposed. In principle it is the simple threshold based vital signs monitoring system designed to identify signs of deteriorating parameters. The proposed track and trigger system (TTS), presented in Figure 7.4, is based on five physiological parameters: respiratory rate, heart rate, temperature, oxygen saturation and systolic blood pressure, each of which is recorded in boxes, according to predefined NEWS ranges. As in any EWS system, points are

allocated to disturbed physiological values, with weightings, to guide intervention or to monitor the effectiveness of interventions. Typically when the score exceed critical values, an alert is triggered, following the NEWS trigger system, that notifies relevant parties. Such system on its own is capable to replace traditional population-based periodic observation charts, where values were recorded and plotted on graphs by hand.

When combined with local inference methods on the personal server the model provides an ability to adjust sensors to achieve the desired level of diagnostic resolution and certainty with respect to individual user. This is achieved through distributed dynamic sensor node model where sensor node continuously monitors values of interest and independently pre-process data by comparing it with the predefined ranges locally. Threshold values can be assigned to individual sensor channels and configured by the sensor node itself base on the settings obtained from the Personal Server. In this settings each sensor node can operate in one of two modes: passive or active.
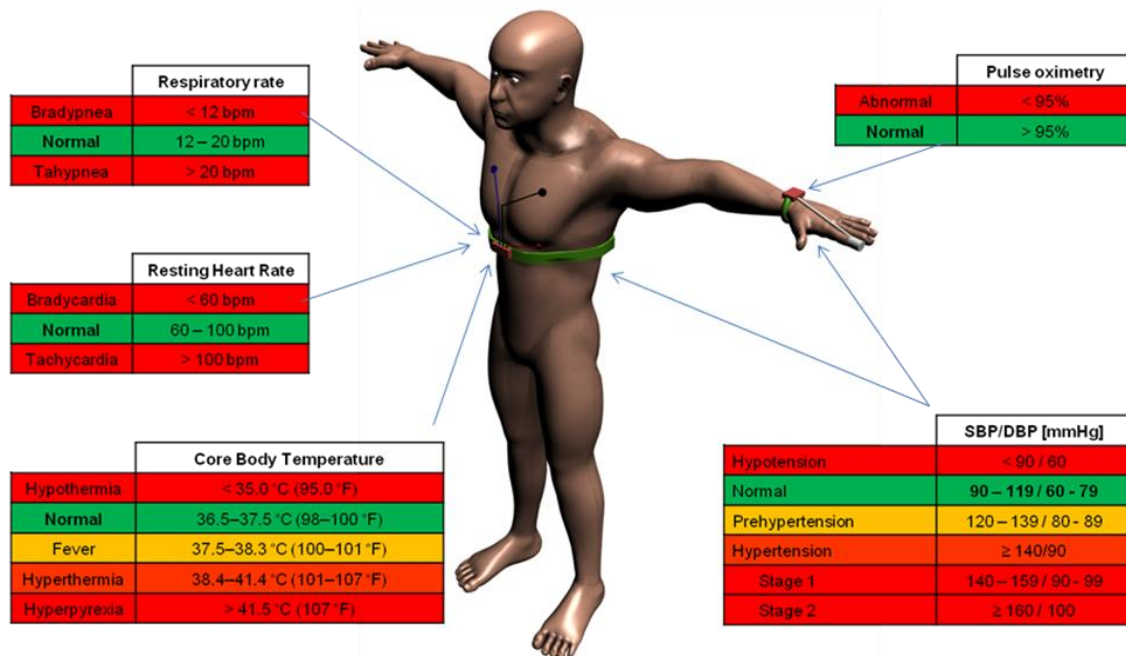
Figure 7.4 1st stage BAN distributed sentinel event triggering model.

Typically when there are no signs of deteriorating parameters (all vital signs are within their optimal ranges), the node operates in passive mode. Passive mode is associated with aggregated score of zero equivalent with green colour in Figure 7.4. In this case all its sensors are continuously sampling and pre-processing digital signals of interest without forwarding them to the Personal Server (PS). It involves signal filtering and conversion of such noise-free signal into numeric vital signs values. In order to enable the adaptive model located on the Personal Server, (presented later in this work) to adjust itself to the patient most optimal normal vital sign ranges, once per minute the sensor node being in passive mode sends to the PS, the mean, median, mode and standard deviation values for each parameter. The Personal Server, using mean analysis, assess the degree of dispersion of numeric physiological data from their normal

ranges and adjust the well-being model to a specific user. This approach allows to minimise data transmission in the network and therefore preserve the battery lifetime.

In turn, when any vital sign is outside of its optimal ranges, the node changes its operation mode to active. This is controlled by the data processing component on the sensor node which validates signals against signal specific sentinel values. These values are used to trigger (or suppress) transmission of all the monitored vital signs numerics in real time to the Personal Server when the predicate condition specified in equation 8 in section 4.4.3.3. Such transmitted signals are further processed in an integrative manner by the PS processing model in order to verify if the trigger of an active mode was a result of an abnormality (true positive) or false alarm (false positive). When all vital signs values return to their normal ranges the node comes back to the passive operation mode.

Such distributed dynamic sensor allocation model, allows to implement the effective user tailored monitoring scheme that is being trained (adjusted to the user) on the Personal Server, while the sensor is in passive mode. This, so called "physiological signature of the individual", also named as "Health Map", is adjusted by aggregation of different features from the collected signals and is used to: 1) asses patients wellness (i.e. how close the given patient is to his/her optimal health status measured in term of vital signs) while the sensor is in passive mode, or 2) as a second reference point to verify/detect any medical abnormalities when any sentinel value on the sensor node was triggered and the node is in the active mode. Such model it therefore able to consider the most optimal individual's personal physiological characteristic in the decision process.

7.5 Continuous signals fiducial markers extraction algorithms

7.5.1 QRS detection algorithm with combined adaptive threshold

One of the crucial steps in real-time computer-based ECG analysis is the accurate detection of the QRS complexes, in particular R wave peaks. The QRS detection is part of the pre-processing stage, responsible for signal segmentation. It extracts the heart-beat like segments out of the continuous ECG signal and input it to the ECG classification algorithm. Apart of the signal segmentation, the QRS complex detection is also important for calculation of the heart rate (HR), which is the number of ventricular depolarisations (i.e. number of QRS complexes) in a minute long period.

Accurate detection of the QRS complexes is difficult to achieve when the monitoring is performed during normal daily activity; mainly due to high noise contamination caused by baseline drifts changes, motion artefacts, muscular noise, or morphological differences in the waveform between subjects (Benitez, Gaydecki, Zaidi *et al.* 2001).

Throughout the literature, several methods have been proposed for the automated detection of QRS complexes in ECG signal. As summarised by Kohler et al. (2002) approaches

to QRS detection are most commonly based on derivatives (Yeh and Wang 2008), digital filters (Pan and Tompkins 1985, Hamilton and Tompkins 1986), linear prediction and wavelet transform (Kadambe, Murray and Boudreaux-Bartels 1999, Martínez, Almeida, Olmos *et al.* 2004, Chen, Chen and Chan 2006, Zidelmal, Amirou, Adnane *et al.* 2012), Hilbert transform and Empirical Mode Decomposition (EMD) (Benitez, Gaydecki, Zaidi *et al.* 2001, Hadj Slimane and Naït-Ali 2010, Manikandan and Soman 2012), geometrical matching(Suarez, Silva, Berthoumieu *et al.* 2007), mathematical morphology (Zhang and Yong 2009), neural networks (Xue, Hu and Tompkins 1992, Hu, Tompkins, Urrusti *et al.* 1994, Cohen, Tompkins, Djohan *et al.* 1995, Abibullaev and Seo 2011), and length/energy transforms (Lee, Jeong, Yoon *et al.* 1996, Paoletti and Marchesi 2006), amongst others. A typical QRS detection algorithm consists of a pre-processing, detection and post-processing stages. The pre-processing stage, if present, in most cases focuses on signal filtering in order to attenuate other signal components and artefacts such as P-wave, T-wave, baseline drift and foremost noise. Frequency components of interest in QRS complex detection range from about 10Hz to about 25Hz (Kohler, Hennig and Orglmeister 2002). Such filtered signal is then used to generate a feature signal by applying various transformations or calculations, in which the occurrence of a QRS is detected according to a specific peak detection logic. Finally, during the post-processing stage, if present, some additional decision rules are most often applied for the reduction of false-positive detections.

The proposed method for QRS detection falls into the last group of algorithms mentioned above, by using the curve-length transform proposed by Paoletti and Marchesi (2006) as a base. What makes this algorithm different is the adaptive signal dependent threshold used as the decision function, as well as other elements during the pre- and post-processing stages. Prior to the actual detection, a band-pass Finite Impulse Response (FIR) filter is applied in order to eliminate undesired noise. For the signal produced by applying the curve-length transform, it is proposed that the required adaptive threshold should be a function of basic mean and standard deviation. Such threshold levels are computed to be signal dependent in a way that an adaptation to changing signal characteristics is possible. Finally, an additional decision rule, based on the average peak-to-peak interval for the reduction of false-positive detections is applied.

7.5.1.1   Curve-length transform

Presented method at its core exploits the curve-length concept (Paoletti and Marchesi 2006), which well characterize the local shape of the signal by looking at how its length changes over a fix period of time. The concept is well illustrated in Figure 7.5 where two examples of ECG signal segments L1 and L2 are presented.
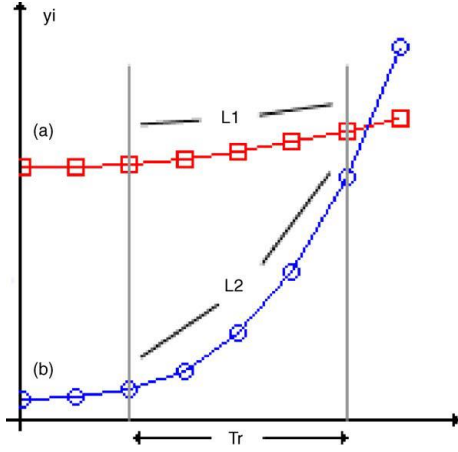
Figure 7.5 The simple curve-length concept. The length of the segment L1 and L2 characterize the local shape of the two signals in time T (Paoletti and Marchesi 2006)

The transformation analyse in time domain the lengths of successive overlapping signal curve segments in a given time interval T. In the discrete time domain the length of the i-th segment can be calculated as:

$$L(i) = \sum_{j=1}^{N_L} \sqrt{T_x^2 + (y_{i+j} - y_{i+j-1})^2} \qquad (34)$$

where T_x is the sampling interval, $(y_{i+j} - y_{i+j-1})$ represents the j-th ECG signal increment, and N is the number of samples in the width of the i-th calculus window.

Assuming that T is a constant in each segment we can eliminate it from the equation and approximate the successive segments $L(i)$ as absolute value of signal increment:

$$L(i) = \sum_{j=1}^{N_L} \sqrt{(y_{i+j} - y_{i+j-1})^2} = \sum_{j=1}^{N_L} |y_{i+j} - y_{i+j-1}| \qquad (35)$$

Since the calculus window is constant for all the overlapping window, the quantity L is larger for ECG segments containing strong slope variation such as QRS complexes, and tends to be smaller for flat segments of slower waves such as P and T waves. To enhance this effect and to make all the output values positive, the nonlinear amplification of the output, by squaring the difference between consecutive points, has been introduced. This presents the following equation.

$$L(i) = \sum_{j=1}^{N_L} (y_{i+j} - y_{i+j-1})^2 \qquad (36)$$

Finally, in order to make the analyses robust against high frequency noise the alternating point's deference instead of consecutive point's differences is calculated as follows:

$$L(i) = \sum_{j=2}^{N_L} (y_{i+j} - y_{i+j-2})^2 \qquad (37)$$

This in practices can be implemented with a low computational cost recursive expression:

$$L(0) = \sum_{j=2}^{N_L} (y_j - y_{j-2})^2$$

$$L(i) = L(i-1) + (y_{i+(N_L/2)} - y_{i+(N_L/2)-2})^2 - (y_{i-(N_L/2)} - y_{i-(N_L/2)+2})$$

(38)

Each time the output is calculated, the new term $(y_{i+(N_L/2)} - y_{i+(N_L/2)-2})^2$ is added to the old value, while the term $(y_{i-(N_L/2)} - y_{i-(N_L/2)+2})^2$ is subtracted. Using the recursive equation (38) each sample of the output value is calculated performing only six operations. This is illustrated in Figure 7.6.
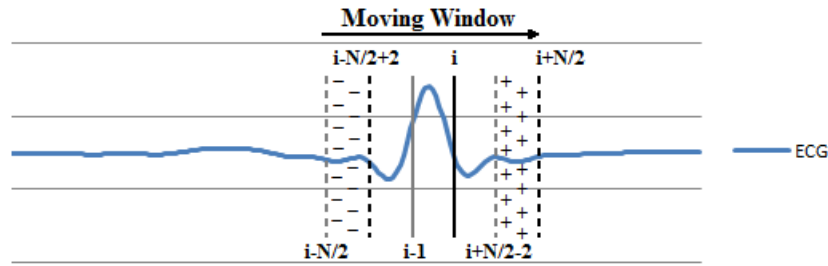
Figure 7.6 Curve-length transforms recursive expression.

The width of the moving window N_L used to calculate the output should be in practice the same as the estimated duration of the episode to be detected; in this case the QRS complexes. Typically, the length of the QRS complex estimates around 100ms and this value is used.

7.5.1.2 Algorithm overview

The block diagram of the proposed QRS detection algorithm is shown in Figure 7.7. As with most QRS detection algorithms, the first stage is formed by a digital filtering components. In our case the band-pass FIR filter with a Hann window is used. Typical frequency components of a QRS complex range from about 10 Hz to about 25 Hz (Kohler, Hennig and Orglmeister 2002). Base on this as well as previous analyses of noise source, the band stop frequencies, were set to 0.5Hz and 25Hz in order to remove baseline wandering and any high frequency noise in the ECG signal. In order to remove a DC offset and adjust peak levels, the signal is normalised to 0-10 mV. The normalization range was established empirically by observing the performance of the algorithm on the training and validation data sets.

In the second stage, is where the curve-length transformation, originally proposed by Paoletti and Marchesi (Paoletti and Marchesi 2006), is applied. As discussed in previous section it characterize the local shape of the signal by using the length of successive (every two points) overlapping ECG signal segments on a given time interval. The operator for this transformation is implemented with a low computational cost as the recursive expression. At this stage the basic mean and standard deviation of the curve-length transform are calculated over the given

N_M and N_S moving windows. Both were established empirically by observing the performance of the algorithm on the training and validation data sets.
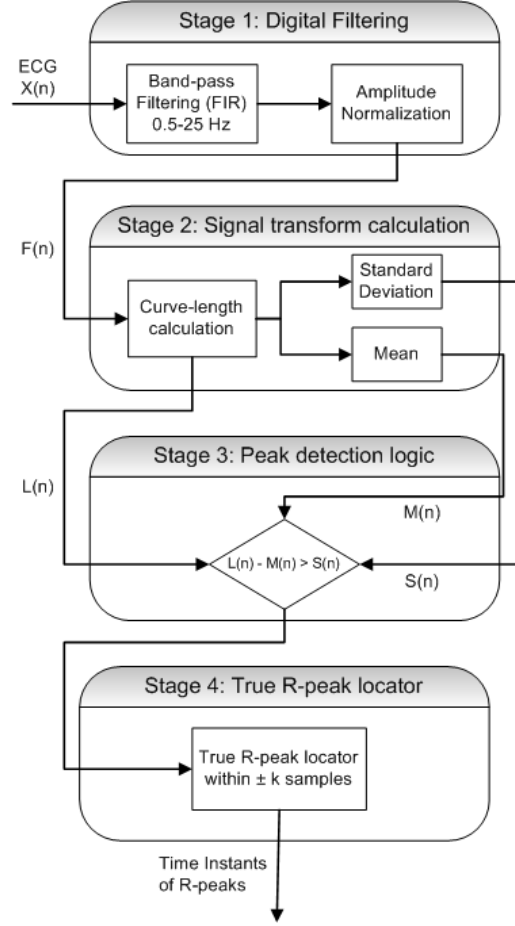
Figure 7.7 Block diagram of the proposed QRS detection algorithm.

The third stage is the peak detection logic based on the curve-length transformation and statistical parameters derived from stage two. The proposed technique originates from a basic statistical dispersion, which measures the spread of a distribution about its mean. It indicates locations of the local maxima by detecting areas of the signal, where the difference between the transformed signal and its mean is larger than the standard deviation. Candidate R-peak points are those which satisfy the condition:

$$L(i) - M(i) > S(i) \tag{39}$$

where $L(i)$ is the transform function, $M(i)$ is the mean function, and $S(i)$ is the standard deviation function. All of them calculated for each ECG point using respectively the moving windows of size N_L, N_M, N_S around that point.

Finally in stage four, a simple post-processing function identifies accurate locations of true R-peaks in the original ECG signal using suggested candidate R-peaks from stage three. It involves retaining maximum peaks with the largest amplitude and removing adjacent peaks that are too close to each other, within the same window of size k. The default value of the

parameter k is set to 300ms. This has been determined empirically based on the average maximum estimated heart rate per individual. The example depicted in Figure 7.8 shows the original ECG signal and its corresponding curve-length, mean and standard deviation functions used to detect the QRS complexes using the proposed four-stage algorithm.
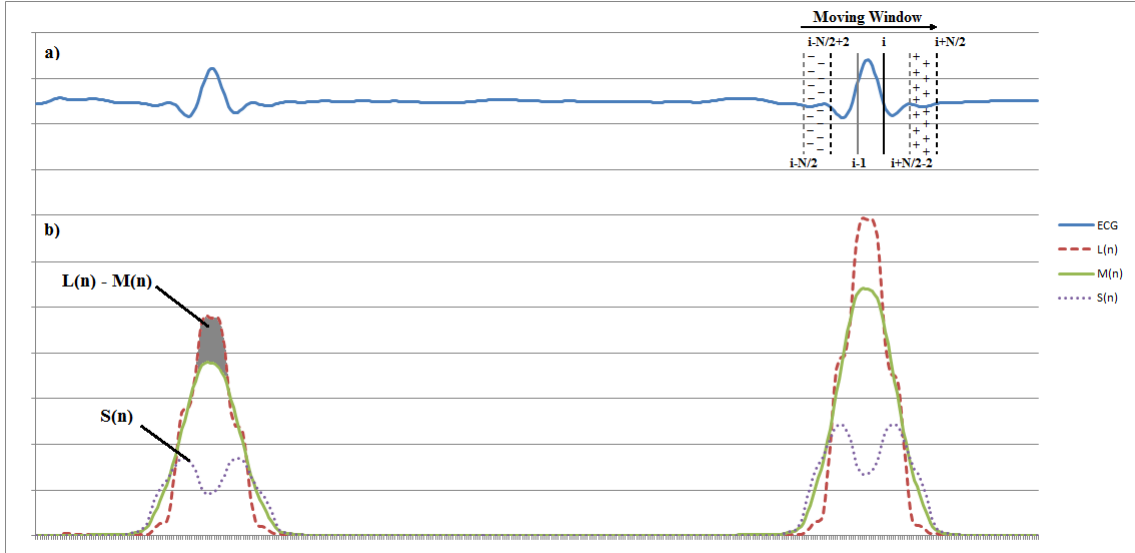
Figure 7.8 Curve-length, mean and standard deviation functions (b) for the given ECG signal (a).
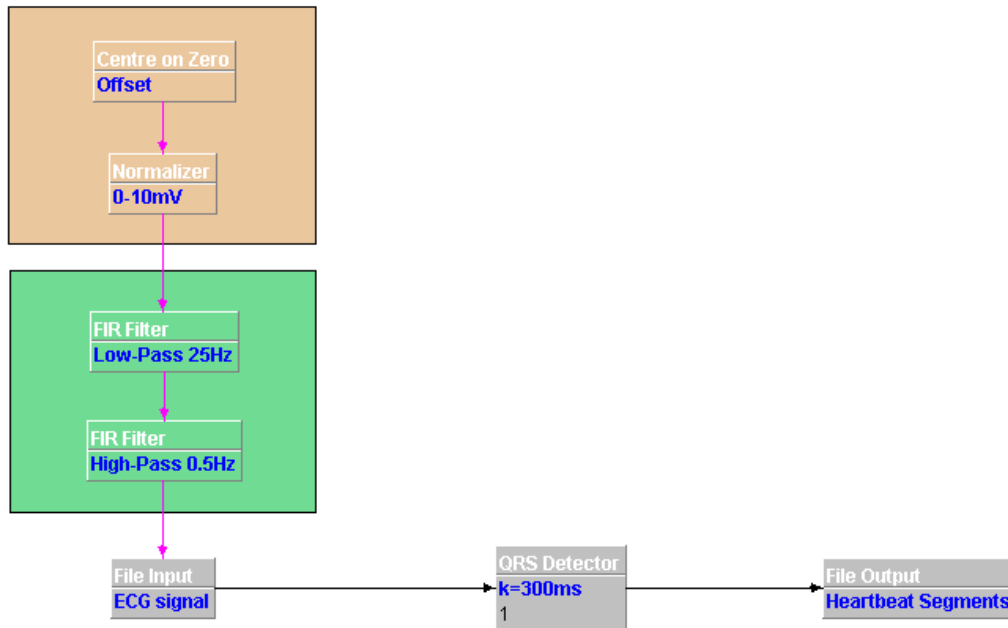
Figure 7.9 Block diagram of QRS detection algorithm in jMENN Editor.

7.5.1.3   Experimental Results

The proposed QRS complex detection algorithm was evaluated using the MIT-BIH arrhythmia database (Goldberger, Amaral, Glass *et al.* 2000). It contains 48 half-hour of two channel ECG recordings obtained from 47 subjects and sampled at 360 Hz with 11-bit resolution over a 10mV range. The subjects were 25 men aged 32 to 89 years, and 22 women

aged 23 to 89 years. Two or more cardiologists independently annotated each record. Any disagreements were resolved to obtain the computer-readable reference annotations for each beat (approximately 110,000 annotations in all) included with the database. The ECG records in this database include signals with acceptable quality, sharp, tall or wide P and T waves, negative QRS complex, muscle noise, baseline drift, QRS morphology variations, multiform premature complex ventricular junctional, and supraventricular arrhythmias, conduction abnormalities, and irregular heart rhythms. All of these signals were included in this study apart from episodes of the ventricular flutter in record 207 which were excluded from the performance analysis for better comparison with the other detection methods (Manikandan and Soman 2012).

In order to evaluate the actual performance of the proposed algorithm, three quantitative results for each test sample were observed according to (American National Standards Institute 1998, Kohler, Hennig and Orglmeister 2002). These are:

- true-positive (TP), defined as the number of actual heart beats, which have been correctly detected (equivalent with hit);
- false-positive (FP), defined as the number of heart beats that are actually not heart beats, but were incorrectly labelled as heart beats (equivalent with false beat);
- and false-negative (FN), defined as the number of heart beats that are actually heart beats but were not labelled as heart beats (equivalent with miss).

To compare the performance of this QRS detection algorithm with other proposed methods, the sensitivity (Se), the positive predictive value (+P) and the failed detection (FD) values were computed using the following equations:

$$Se = \frac{TP}{TP + FN} \times 100\% \tag{40}$$

$$+P = \frac{TP}{TP + FP} \times 100\% \tag{41}$$

$$FD = \frac{FN + FP}{TP + FN} \times 100\% \tag{42}$$

The overall performance of the method is measured in terms of the detection accuracy. While it is understood that accuracy is commonly evaluated as:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \tag{43}$$

Because the QRS detection is the unary problem, that yield only an impulse when the peak is detected, the True Negative (TN) results are difficult to name and quantify with the continuous signal. Therefore in order to be consistent with literature on QRS detection and keep the results comparable with similar studies (Pan and Tompkins 1985, Paoletti and Marchesi 2006, Zhang

and Yong 2009, Adnane, Jiang and Choi 2009, Manikandan and Soman 2012) the formula used to calculate overall accuracy is defined as:

$$Acc = \frac{TP}{TP + FP + FN} \times 100\%$$

(44)

Table 7.8 summarises the results obtained during the tests on the first-channel of 48 ECG recordings of the MIT-BIH arrhythmia database. The proposed algorithm produced, in total 135 false negatives (FN), and 148 false positives (FP) resulting in a total detection failure of 283 beats. This resulted in the overall failed detection rate of 0.26%, sensitivity at the level of 99.88%, and positive predictive value at the level of 99.86%. The overall detection accuracy achieved by the algorithm was 99.74% for all annotated beats. Looking at the individual ECG records the detection accuracies vary from 98.53% to 100%, sensitivity vary from 99.13% to 100%, positive predictive value vary from 99.18% to 100% and failed detection vary from 0.00% to 1.48% depending on the characteristics of ECG signal morphology as well as the noise.

The analyses of the original ECG signal revealed that, records 105 and 203 are the most difficult ones for analysis. Both of them contain an exceptionally large amount of high-grade noise and artefacts when compared to the other rather moderate noise contaminated records of 104, 108, 200, 210, and 228, Both of them are also affected by severe baseline drifts and abrupt changes that are also present in records 108, 111, 112, 116, 201 which had a significant impact on the number of false positives produced on these records. Record 203 additionally exhibits various irregular rhythms (also present in records 201, 202, 219 and 222), multiform ventricular arrhythmia, negative QRS polarity and sudden changes in QRS morphology (also present in records 200 and 233). This had a significant impact on the number of false negatives. Therefore these two records produce the largest number of false detections, not only in the obtained results, but also throughout all the reviewed algorithms. However, as can be observed from table 7.8, the performance of the proposed algorithm for all the remaining records, except for records 116 and 108, from these three groups, was above the average. The large number of false negatives in record 116, as well as in 208, is caused by smaller QRS complexes than in the other records. Record 108 contains tall sharp P waves which makes the algorithm produce larger than average number of false positives for this record. Nevertheless, results presented here compare favourably with published results of other QRS detection algorithms, as evidenced when comparing the proposed algorithm with other methods in Section 8.3.1. It should be noted that results presented in table 7.8 differ from what has been reported in (Lewandowski, Arochena, Naguib *et al.* 2012) due to window size parameters N_M, N_s which in this version of the algorithm were increased to $N_M = N_s = 100$, what improved the results.

Record	Total(beats)	TP(beats)	FN(beats)	FP(beats)	FD(beats)	Se(%)	+P(%)	FD(%)	Acc(%)
100	2273	2273	0	0	0	100.00	100.00	0.00	100.00
101	1865	1863	2	5	7	99.89	99.73	0.38	99.63
102	2187	2187	0	0	0	100.00	100.00	0.00	100.00
103	2084	2084	0	0	0	100.00	100.00	0.00	100.00
104	2229	2223	6	5	11	99.73	99.78	0.49	99.51
105	2572	2555	17	21	38	99.34	99.18	1.48	98.53
106	2027	2027	0	4	4	100.00	99.80	0.20	99.80
107	2137	2137	0	0	0	100.00	100.00	0.00	100.00
108	1763	1753	10	8	18	99.43	99.55	1.02	98.98
109	2532	2530	2	0	2	99.92	100.00	0.08	99.92
111	2124	2123	1	3	4	99.95	99.86	0.19	99.81
112	2539	2539	0	1	1	100.00	99.96	0.04	99.96
113	1795	1794	1	2	3	99.94	99.89	0.17	99.83
114	1879	1879	0	4	4	100.00	99.79	0.21	99.79
115	1953	1953	0	0	0	100.00	100.00	0.00	100.00
116	2412	2391	21	2	23	99.13	99.92	0.95	99.05
117	1535	1535	0	1	1	100.00	99.93	0.07	99.93
118	2278	2278	0	2	2	100.00	99.91	0.09	99.91
119	1987	1987	0	0	0	100.00	100.00	0.00	100.00
121	1863	1862	1	2	3	99.95	99.89	0.16	99.84
122	2476	2476	0	0	0	100.00	100.00	0.00	100.00
123	1518	1518	0	1	1	100.00	99.93	0.07	99.93
124	1619	1619	0	1	1	100.00	99.94	0.06	99.94
200	2601	2597	4	5	9	99.85	99.81	0.35	99.65
201	1963	1956	7	0	7	99.64	100.00	0.36	99.64
202	2136	2135	1	2	3	99.95	99.91	0.14	99.86
203	2980	2958	22	18	40	99.26	99.40	1.34	98.67
205	2656	2651	5	0	5	99.81	100.00	0.19	99.81
207	1860	1857	3	8	11	99.84	99.57	0.59	99.41
208	2955	2941	14	4	18	99.53	99.86	0.61	99.39
209	3005	3005	0	7	7	100.00	99.77	0.23	99.77
210	2650	2648	2	7	9	99.92	99.74	0.34	99.66
212	2748	2748	0	0	0	100.00	100.00	0.00	100.00
213	3251	3250	1	0	1	99.97	100.00	0.03	99.97
214	2262	2259	3	3	6	99.87	99.87	0.27	99.74
215	3363	3362	1	0	1	99.97	100.00	0.03	99.97
217	2208	2206	1	2	3	99.95	99.91	0.14	99.86
219	2154	2154	0	0	0	100.00	100.00	0.00	100.00
220	2048	2048	0	1	1	100.00	99.95	0.05	99.95
221	2427	2427	0	0	0	100.00	100.00	0.00	100.00
222	2483	2480	3	8	11	99.88	99.68	0.44	99.56
223	2605	2604	1	0	1	99.96	100.00	0.04	99.96
228	2053	2048	5	9	14	99.76	99.56	0.68	99.32
230	2256	2256	0	2	2	100.00	99.91	0.09	99.91
231	1571	1571	0	0	0	100.00	100.00	0.00	100.00
232	1780	1780	0	10	10	100.00	99.44	0.56	99.44
233	3079	3078	1	0	1	99.97	100.00	0.03	99.97
234	2753	2753	0	0	0	100.00	100.00	0.00	100.00
Total	109494	109358	135	148	283	99.88	99.86	0.26	99.74

Table 7.8 Results of performance evaluation for the proposed QRS detection algorithm using MIT/BIH Database.

7.6 Physiological signals analysis using supervised learning algorithms

Many different bio-signals can be detected on the surface of the human body. Types of bio-signals range from most common electrical to non-electrical signals such as mechanical, acoustic, chemical or optical. Many of such signals help to reflect the internal status thus providing information on condition of the whole body as well as particular internal organs. Among this group the organ which is of particular importance to humans health is the heart. This is due to its crucial life support function. Its function can be monitored in many different ways using multiple measures and indexes. The most basic one commonly used every day, focuses on numerical analyses of the heart rate. Signal processing methods, which can support this analysis was presented in section 7.5.1.

Simple numerical and statistical analyses of the heart rate, such as Heart Rate Variability, despite being a powerful tool for assessing the functioning of the whole cardiovascular system, provides only a fractional information on condition of heart itself. Instead, to provide diagnostically significant information we used the electrocardiogram, which is a graphic recording of the electrical activities in human heart. Its shape, size and duration reflect the heart rhythm over time. The varied sources of heart diseases provide a wide range of alterations in the shape of the electrocardiogram called heart arrhythmias. Heart arrhythmia is any abnormal cardiac beat and/or rhythm. It result from any disturbance in the rate, regularity and site of origin or conduction of the cardiac electric impulse (Thaler 2007). Classification of arrhythmia is an important step in developing devices for monitoring the health of individuals.

7.6.1 Binary ECG abnormalities detection using Support Vector Machine

In general, arrhythmias can be divided into two groups. The first group includes ventricular fibrillation and tachycardia, which are life threatening and require immediate therapy with a defibrillator. The second group includes arrhythmias that are not imminently life-threatening but may require therapy to prevent development of further heart disease conditions (De Chazal, O'dwyer and Reilly 2004). Many arrhythmias manifest as sequence of heartbeats with unusual timing or morphology. Therefore an important step towards arrhythmia detection is the classification of shape of the heartbeats as well as rhythm between consecutive classified heartbeats in the signal.

According to Mele (2008) an estimated 300 million of ECG recordings are performed every year. To support their analysis and interpretation many automated computer-based methods have been proposed. However, automated classification of ECG beats is a challenging problem as the morphological and temporal characteristics of ECG signals show significant variations for different patients and under different temporal and physical conditions (Hoekema, Uijen and Van Oosterom 2001). This is the reason in practice for the underperformance of many fully automated ECG processing systems, which hence make them unreliable to be widely used

clinically (De Chazal and Reilly 2006). With more and more Smart Wearable Systems for physiological signals monitoring, there is a need for more accurate, reliable and efficient automated ECG interpretation algorithms, which will be capable of detecting cardiac abnormalities for both real-time clinical as well as well-being monitoring applications. Ubiquities arrhythmia detection could significantly improve cardiovascular diseases diagnosis and treatment (Sneha and Varshney 2009).

Throughout the literature, variety of ECG features have been proposed for the automated classification of ECG signal (Karpagachelvi, Arthanari and Sivakumar 2010). These include ECG morphology (Tadejko and Rakowski 2007) and heart beat interval features (De Chazal, O'dwyer and Reilly 2004, Christov, Gómez-Herrero, Krasteva *et al.* 2006), frequency-domain features (Lin 2008, Dutta, Chatterjee and Munshi 2010), higher order cumulant features (Osowski and Linh 2001), Hermite polynomials (Lagerholm, Peterson, Braccini *et al.* 2000, Linh, Osowski and Stodolski 2003) or Karhunen-Loeve transform (Yu-Hen Hu, S. Palreddy and Willis J. Tompkins 1997, Paoletti and Marchesi 2006) amongst others. However, the most popular ECG feature extraction algorithm is wavelet transformation (WT), which was used by (Sahambi, Tandon and Bhatt 1997, Martínez, Almeida, Olmos *et al.* 2004, Inan, Giovangrandi and Kovacs 2006, Özbay 2009, Daamouche, Hamami, Alajlan *et al.* 2011, Kim, Min and Lee 2011), to name a few. This is because ECG signal is a highly non-stationary signal, and the inherent properties of WT include good time-frequency location and cross sub-band similarity (Daubechies 1990). Finally as authors in (Guyon and Elisseeff 2003) suggest, it is of significant importance to select suitable features from all WT coefficients, as well as their statistical values for efficient classification of extremely similar classes with lower computation complexity.

Other than feature extraction, the performance of a recognition system highly depends on the type of the classifier. Amongst the literature various classification rules were applied to ECG arrhythmia detection including: linear discriminants (Jekova, Bortolan and Christov 2004, De Chazal, O'dwyer and Reilly 2004), artificial neural networks (ANN) (Dokur and Ölmez 2001, Güler and Übeylı˙ 2005, Hosseini, Luo and Reynolds 2006, Özbay and Tezel 2010, Özbay, Ceylan and Karlik 2011), fuzzy logic methods (Engin 2004, Özbay, Ceylan and Karlik 2006), Hidden Markov Model (HMM) (Hughes, Tarassenko and Roberts 2004, Andreao, Dorizzi and Boudy 2006, Gomes, Soares, Correia *et al.* 2010), self-organizing maps (SOM) (Lagerholm, Peterson, Braccini *et al.* 2000, Gacek 2011), or mixture of experts (MOE) (Y. H. Hu, S. Palreddy and W. J. Tompkins 1997, Javadi, Arani, Sajedin *et al.* 2013) and hybrid classifiers (Yaghouby, Ayatollahi, Bahramali *et al.* 2010, Homaeinezhad, Atyabi, Tavakkoli *et al.* 2012, Doğan and Korürek 2012) amongst other methods.

All these algorithms fall in to one of three main groups of ECG classifiers, according to number of arrhythmia classes that they can distinguish. Hence we have binary and multiclass algorithms. Amongst the binary classifiers we can further distinguish those algorithms which in general classify heartbeats between normal and abnormal beats or those algorithms which look for specific abnormalities in the ECG signal such as premature ventricular contractions (PVC)

(Sayadi, Shamsollahi and Clifford 2010) of ischemic beats (Goletsis, Papaloukas, Fotiadis *et al.* 2004). The multiclass classifiers group further splits on multiclass classification of up to 7 ECG beats or more. Because this research project focus on real-time ubiquities vital signs abnormalities detection using small wearable sensor devices with limited processing power and battery life, therefore our focus in the remaining sections is on the binary classification between normal and abnormal beats only.

Natural choice for such defined problem are those algorithms which are binary by definition. The SVM algorithm, is a typical binary classifier, which determines the hyper-plane, which is the decision boundary in a high dimension feature space, enabling to distinguish between members of two classes. This higher-dimensional space is called the feature space, as opposed to the input space occupied by the training examples. With an appropriately chosen feature space of sufficient dimensionality, any consistent training set can be made separable. What characterise SVM is their ability to produce very accurate classifiers which avoid over fitting (a problem of finding trivial solutions that over fit the data) by choosing a specific hyper plane that maximizes the minimum distance from the hyperplane to the closest training point. Moreover, the computational complexity of the classification operation does not depend on the dimensionality of the feature space, which can even be infinite. Next, the separating hyperplane is represented sparsely as a linear combination of points. The system automatically identifies a subset of informative points and uses them to represent the solution. Finally, the training algorithm solves a simple convex optimization problem. All these features make SVMs an attractive classification system (Brown, Grundy, Lin *et al.* 1999). As noted by Shen et al. (Shen, Kao, Yang *et al.* 2012) these alternative approach has recently attracted researchers such as (Melgani and Bazi 2008, Kampouraki, Manis and Nikou 2009, Moavenian and Khorrami 2010) to study the feasibility of applying it to ECG analysis problem. As further reported by these authors, the overall results that can be obtained with SVM are superior to other algorithms, however the error rate is still higher than 10% for a few specific classes.

Following this analyses we have proposed a new method for cardiac arrhythmia detection which uses the support vector machine with wavelet transform (WT) feature extraction. For this purposes the level 3 Daubeches (db4) discrete wavelet transform (DWT) is used. The DWT analyses the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. For this purpose the DWT employs two sets of functions, called scaling functions and wavelet functions, which are associated with low-pass and high-pass filters, respectively. Such coarse approximation of the ECG beat at level 3 (extracted using the QRS detection algorithm presented in section 7.5.1) as well as various types of statistical values for each frequency sub-band and the beat, are used as the input vector to the classifier. The classifier is implemented as the SVM with Radial Basis Function (RBF) kernel, which performs the binary classification the normal and abnormal beat classes according to ANSI/AAMI EC57:2012 standard (Association for the Advancement of Medical Instrumentation 2012).

7.6.1.1 ECG arrhythmia classification

The ANSI/AAMI EC57:2012 (Association for the Advancement of Medical Instrumentation 2012) standard recommends five beat classes i.e. normal beat (N), supraventricular ectopic beat (S), ventricular ectopic beat (V), fusion of ventricular and normal beat (F) and unknown beat (Q). In agreement with the AAMI recommended practice the data from the MIT-BIH arrhythmia database (Moody and Mark 2001) was used in this study to represent these five classes of beats during the training and testing of the classifier. The database contains 48 recordings, of many common and life-threatening arrhythmias along with examples of normal sinus rhythms, each containing two 30 minutes long ECG lead signals denoted as lead A and B. In 45 recordings, lead A is a modified-lead II and for the other three is lead V5 of the Einthoven lead system (Barill 2005). Lead B in turn is the V1 lead for 40 records and either lead II, V2, V4 or V5 for remaining records. The first 23 records intends to serve as a representative sample of the variety of waveforms and artefact that an arrhythmia detector might encounter in routine clinical use. Remaining 25 records were chosen to include complex ventricular, junctional, and supraventricular arrhythmias and conduction abnormalities.

There are over 109,000 labelled beats, which form 15 different specific heartbeat types according to MIT-BIH adopted classification. According to the AAMI recommendations the four recordings containing paced beats (/) and fusion of paced and normal beat (f) were removed from the analyses as records with paced beats do not retain sufficient signal quality for reliable processing. The original ECG arrhythmia waveforms corresponding to remaining 13 types of MIT-BIH arrhythmia beats are plotted in Figure 7.10. In order to have a consistent classification model and be able to compare obtained results with other works in the field of ECG classification the AAMI recommended practice was used to combine the MIT-BIH heartbeat types into five main heartbeat classes as shown in Table 7.9. The resulting five classes were further combined to two classes for binary classification to Normal and Abnormal beats, represented as binary 1 and 0 respectively.

AAMI heartbeat class	N	S	V	F	Q
Description	Normal beat	Supraventricular ectopic beat	Ventricular ectopic beat	Fusion beat	Unknown beat
MIT-BIH heartbeat types	Normal beat (N)	Atrial premature beat (A)	Premature ventricular contraction (V)	Fusion of ventricular and normal beat (F)	Paced beat (/)
	Left bundle branch block beat (L)	Aberrated atrial premature beat (a)	Ventricular escape beat (E)		Fusion of paced and normal beat (f)
	Right bundle branch block beat (R)	Nodal (junctional) premature beat (J)			Unclassifiable beat (Q)
	Atrial escape beat (e)	Supraventricular premature beat (S)			
	Nodal (junctional) escape beat (j)				
Normal/Abnormal	**Normal**	**Abnormal**			

Table 7.9 Mapping of the MIT-BIH Arrhythmia Database heartbeat types to the AAMI heartbeat classes
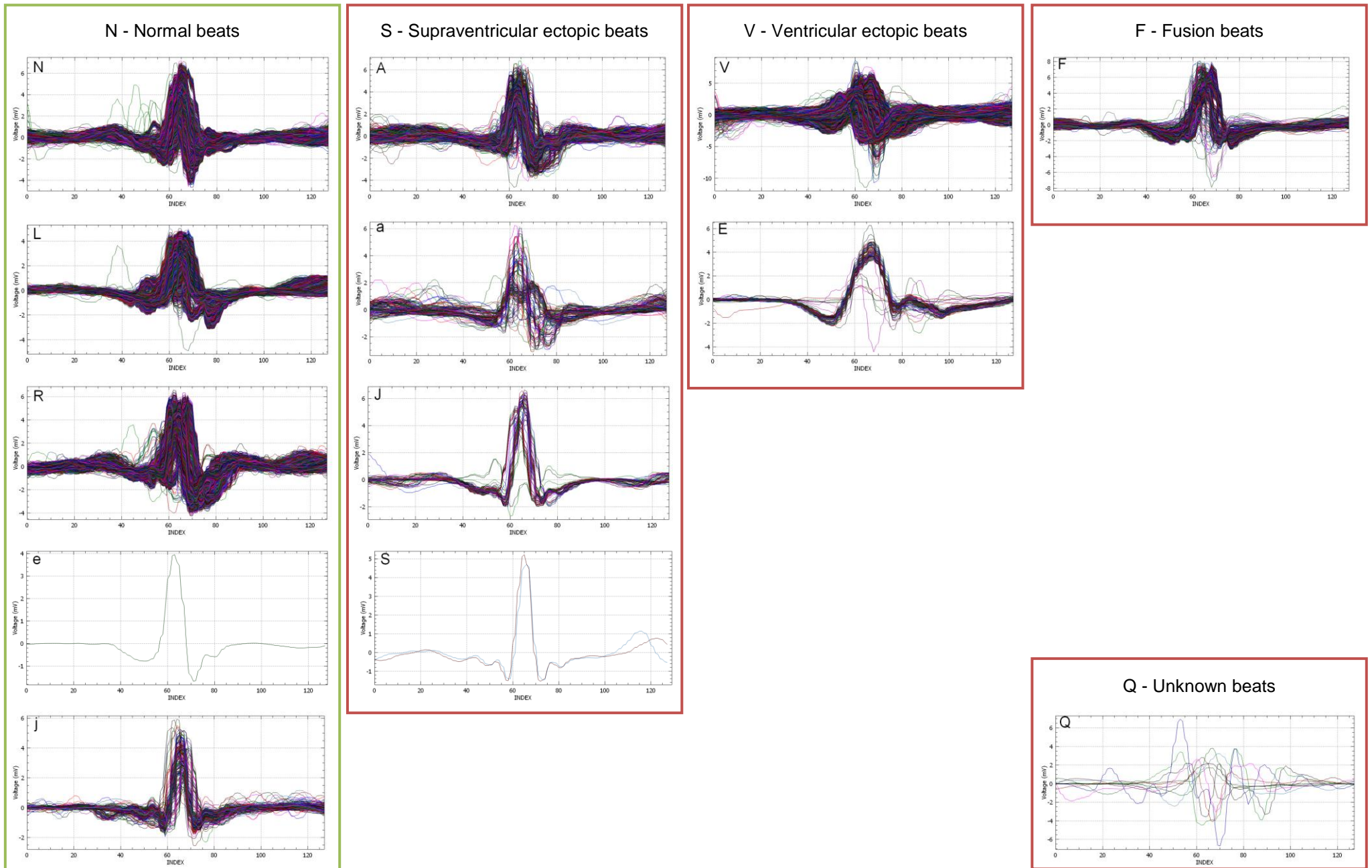
Figure 7.10 The original 13 different MIT-BIH ECG beats mapped to five types of arrhythmia beats according to AAMI classification.

For training, testing and validation purposes the MIT-BIH arrhythmia database ECG recordings were divided, according to De Chazal et al. (De Chazal, O'dwyer and Reilly 2004) into two datasets with each dataset containing ECG data from 22 recordings with the same approximate proportion of beat types. The first dataset (DS1) is used for classifier model selection using cross-validation scheme. One round of cross-validation involves partitioning a DS1 (or representative part of it) into complementary subsets, performing the analysis on one subset (called the training set), and validating the analysis on the other subset (called the validation set). To reduce variability, multiple rounds of cross-validation are performed using different partitions, and the validation results are averaged over the rounds. Finally, when the best model is known the whole DS1 set is used to train the final classifier. The second dataset (DS2) is used for a final performance evaluation and comparison with other algorithms.
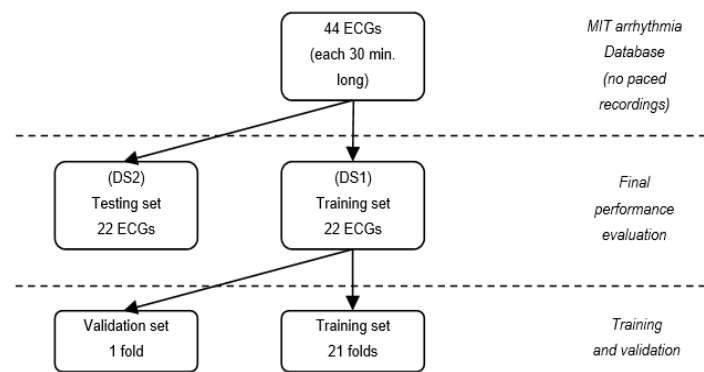
Figure 7.11 Division of the MIT-BIH arrhythmia database into training and testing sets.

Both datasets contain approximately 50 000 heartbeats and have a mixture of the routine and complex arrhythmia recordings. Table 7.10 shows the breakdown of each dataset by heartbeat type. It can be noted from the table below, that there is a large difference in number of examples of the heartbeat types. The largest class is normal beat (N) with almost 75 000 examples while the smallest class is supraventricular premature beat (S) with only two samples. It should be noted that beats segmentation and therefore this breakdown was obtained by executing the QRS detection algorithm, presented in section 7.5.1, on whole MIT-BIH database. Due to QRS algorithm accuracy, numbers in this table differ from the original manually annotated MIT-BIH Arrhythmia table of beat types, which can be found here (Moody 1997).

Heartbeat type		N	L	R	e	j	A	a	J	S	V	E	F	Q	
Heartbeat class				N				S			V		F	Q	Total
Full database	number	74468	8070	7258	16	229	2543	145	83	2	6876	106	799	10	100605
	% of total	74.02	8.02	7.21	0.02	0.23	2.53	0.14	0.08	0.00	6.83	0.11	0.79	0.01	100.0
	records	37	4	6	1	5	27	7	5	1	33	2	17	5	
DS1 (Records: 101, 106, 108, 109, 112, 114, 115, 116, 118, 119, 122, 124 ,201, 203, 205, 207, 208, 209, 215, 220, 223, 230)	number	38049	3947	3782	16	16	809	96	32	2	3663	105	412	4	50933
	% of total	74.70	7.75	7.43	0.03	0.03	1.59	0.19	0.06	0.02	7.19	0.21	0.81	0.01	100.0
	records	18	2	3	1	3	14	3	3	1	17	1	10	3	
DS2 (Records: 100, 103, 105, 111, 113, 117, 121, 123, 200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233, 234)	number	36419	4123	3476	0	213	1734	49	51	0	3213	1	387	6	49672
	% of total	73.32	8.30	7.00	0.00	0.43	3.49	0.10	0.10	0.00	6.47	0.00	0.78	0.01	100.0
	records	19	2	3	0	2	13	4	2	0	16	1	7	2	

Table 7.10 Heartbeat types associated with the extracted beats using our QRS detection algorithm for the Full Database, Dataset 1 (DS1) and Dataset 2 (DS2) from the MIT-BIH Arrhythmia Database.

7.6.1.2 ECG features extraction

It is observed in the literature that ECG feature extraction methods can be divided into three functional groups: direct methods, transformation methods and characteristic parameter methods (Dokur and Ölmez 2001). Direct methods relay on heartbeat magnitudes analysis. In most studies of this group, dimension reduction methods are used to simplifier the classifier structure. Since future vectors are only formed by the magnitudes of the ECG signal, the computation cost is low. However, such futures vectors are affected on one side by the R peak detection, which can scatter vectors in the feature space, and on the other side by the signal noise, which directly affects the elements of the feature vector.

Transformation methods transform original samples to new domain in which classification is performed. It allows to obtain further information from the signal that is not readily available in the raw signal. It prevents the scattering of the vector in the future space but at a cost of increased computational load. Common transformations are Wavelet transform or Fourier transform amongst others.

Finally the last group of ECG feature extraction methods are characteristic parameter estimation methods. Most commonly methods in this group determine the fiducial points in the signal like R, Q, R S or T as well as other statistical or morphological higher level indicators. Such futures vectors are formed often by time intervals of characteristics points and/or signal magnitudes at these points. This group of feature vectors characterise normally by the low dimension however can be affected by various noise sources but most often by variations between different patients.

Proposed ECG classification method use the combination of two feature extraction methods, which belong to the last two groups. These are Daubechies D4 wavelet transform, used by the classifier to analyse heart beat morphology, and the measure of dispersion of local RR interval ratios, which is used to represent the rhythm inherent characteristic of each heartbeat. Such features vector allows to capture all relevant time and frequency domain information for the successful coarse grain classification into normal or abnormal beats.

7.6.1.2.1 Beat analysis by Daubechies D4 wavelet transform

The discrete wavelet transform (DWT) is a linear operation that decomposes a signal into components that appear at different scales. It is used as a measure of similarity which describe the level of correlation between a wavelet at different scales and the signal with the scale (or the frequency). A time-scale representation of a digital signal is obtained using series of digital filtering techniques, where filters of different cut-off frequencies are used to analyse the signal at different scales. Such filtering operations change both the scale of the signal and the resolution which is a measure of the amount of detail information in the signal. Let's consider a discrete signal $x[n]$, being passed through a half band digital low pass filter with impulse response $h[n]$. Filtering a signal corresponds to the mathematical operation of convolution of the signal with the impulse response of the filter defined by equation (45).

$$y[n] = x[n] \times h[n] = \sum_{k=-\infty}^{\infty} x[k] \cdot h[n-k] \qquad (45)$$

A half band low pass filter removes all frequencies that are above half of the highest frequency in the signal. For example, if a signal has a maximum of 200 Hz component, then half band low pass filtering removes all the frequencies above 100 Hz. This can be interpreted as losing half of the information. Therefore half of the samples can be eliminated according to the Nyquist's rule since the highest frequency of the resulting signal has a half of its original highest frequency, what makes half of the number of samples redundant. This is called sub-sampling or down-sampling operation. Sub-sampling a signal corresponds to reducing the sampling rate, or removing some of the samples of the signal. For example, sub-sampling by two refers to dropping every other sample of the signal. Sub-sampling by a factor n reduces the number of samples in the signal n times. This procedure mathematically be expressed as in equation (46).

$$y[n] = \sum_{k=-\infty}^{\infty} h[k] \cdot x[2n-k] \qquad (46)$$

The DWT use the above filtering principle to analyses the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. DWT employs two sets of functions, called scaling functions and wavelet functions, which are associated with low pass and high pass filters, respectively. The decomposition of the signal into different frequency bands is simply obtained by successive high pass and low pass filtering of the time domain signal. The original signal $x[n]$ is first passed through a half band high pass filter $g[n]$ and a low pass filter $h[n]$ and then sub-sampled by 2, simply by discarding every other sample. This constitutes one level of decomposition and can mathematically be expressed as follows:

$$y_{high}[k] = \sum_{n} x[n] \cdot g[2k-n]$$
$$y_{low}[k] = \sum_{n} x[n] \cdot h[2k-n] \qquad (47)$$

where $y_{high}[k]$ and $y_{low}[k]$ are the outputs of the high pass and low pass filters, respectively, after sub-sampling by 2. At every level, the filtering and sub-sampling will result in half the number of samples (and hence half the time resolution) and half the frequency band spanned (and hence double the frequency resolution). For example, suppose we have a 128-sample long signal sampled at 200 Hz and we wish to obtain its DWT coefficients. According to Nyquist's rule the highest frequency component that exists in the signal is 100 Hz. At the first level, the signal is passed through the low pass filter $h[n]$, and the high pass filter $g[n]$, the outputs of which are sub-sampled by two. The high pass filter output is the first level DWT coefficients. There are 64 of them, and they represent the signal in the 50-100 Hz range. These 64 samples are the last 64 samples plotted. The low pass filter output, which also has 64

samples, but spanning the frequency band of 0-50 Hz, are further decomposed by passing them through the same $h[n]$ and $g[n]$. The output of the second high pass filter is the level 2 DWT coefficients and these 32 samples precede the 64 level 1 coefficients in the plot. The output of the second low pass filter is further decomposed, to obtain the level 3 DWT coefficients of size 16. The above procedure, which is also known as the sub-band coding, can be repeated for further decomposition as illustrated in Figure 7.12.
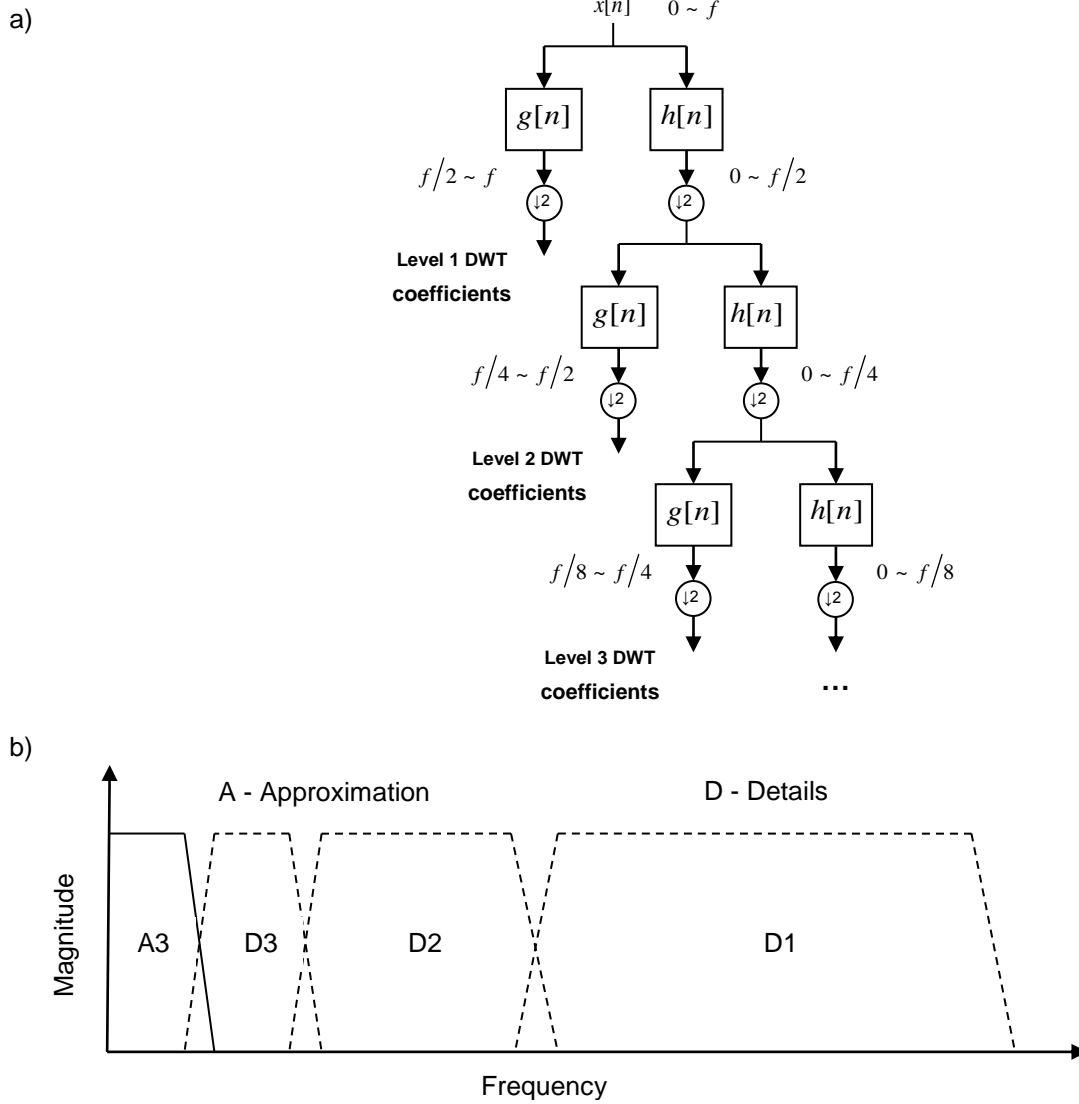
Figure 7.12 a) The DWT sub-band coding algorithm; b) Frequency allocation after a 3-level DWT.

The above diagram is also called the filter bank, where low and high pass filter coefficients $h[n]$ and $g[n]$ are determined by the choice of wavelet function that best match the application. For ECG analysis the natural choice was the family of Daubechies wavelets known for being robust and fast for identifying signals with both time and frequency characteristic (Fugal 2009). This is the family of orthogonal wavelets defining a discrete wavelet

transform, that can is derived by scaling (or dilating) by a factor of a and translating (or shifting) by a factor of b, a mother wavelet function given by equation (48).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \tag{48}$$

Associated to this oscillating (high pass) wavelet function ψ is a non-oscillating (low pass) scaling function ϕ also called father wavelet. In general, a scaling function, is the solution to a dilation equation of the form:

$$\phi(t) = \sum_{k=-\infty}^{\infty} a_k \phi(St - k) \tag{49}$$

where S is a scaling factor (usually chosen as 2) and a_k are constant coefficients, called filter coefficients for which it is often the case that only a finite number of these are non-zero. From the practical point of view the fast wavelet transform algorithm does not make use of the wavelet and scaling functions, but of the filters h and g which characterize the interaction between ψ and ϕ as follows:

$$\phi(t) = \sum_n h[n]\phi[2t - n]$$
$$\psi(t) = \sum_n g[n]\phi[2t - n] \tag{50}$$

The low pass filter coefficients can be derived by imposing certain conditions on the scaling function, what is further discussed by Williams and Amaratunga (Williams and Amaratunga 1994). The high pass filter coefficients g in turn is computed directly from the low pass filter as:

$$g[n] = (-1)^{1-n} h[1-n] \tag{51}$$

The simplest filters that can be derived this way are the Haar filters with only two wavelet and two scaling function coefficients. Daubechies wavelet transform extends the Haar wavelets by using longer filters, that produce smoother scaling functions and wavelets. Furthermore, what characterise Daubechies wavelets is a the maximum number of vanishing moments for a given filter length. The rule is that the larger the size $p = 2k$ of the filter the higher is the number k of vanishing moments. A high number of vanishing moments allows to better compress regular parts of the signal. However, increasing the number of vanishing moments also increases the size of the wavelets, which can be problematic in singular part where the signal for instance discontinuous. Therefore choosing the best wavelet, and thus choosing k, that is adapted to a given class of signals, thus corresponds to a trade-off between efficiency in regular and singular parts. As such the filter with $k = 1$ vanishing moments corresponds to Haar filter or Daubechies D2 wavelet (D2 denotes number of coefficients); the filter with $k = 2$ vanishing moments corresponds to Daubechies D4 which compress perfectly the linear signals,

while the filter with $k = 3$ vanishing moments (Daubechies D6), compress perfectly the quadratic signals. This explains why the db4 wavelet transform was selected for this application.

The Daubechies D4 transform has four wavelet and four scaling function coefficients. The scaling function coefficients are:

$$h_0 = \frac{1+\sqrt{3}}{4\sqrt{2}}; \qquad h_1 = \frac{3+\sqrt{3}}{4\sqrt{2}}; \qquad h_2 = \frac{3-\sqrt{3}}{4\sqrt{2}} \qquad h_3 = \frac{1-\sqrt{3}}{4\sqrt{2}} \tag{52}$$

, while the corresponding wavelet function coefficient values are:

$$g_0 = h_3; \qquad g_1 = -h_2; \qquad g_2 = h_1; \qquad g_3 = -h_0 \tag{53}$$

Figure 7.13 plots the scaling and wavelet functions along with their Fourier coefficient amplitudes.
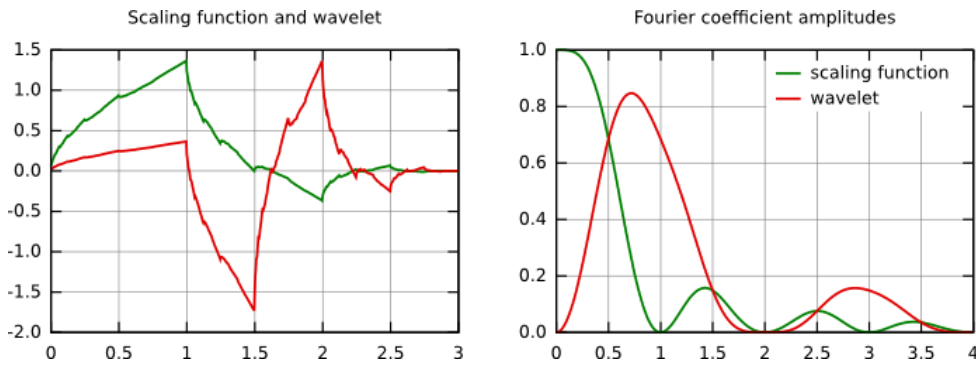
Figure 7.13 Daubechies D4 scaling and wavelet functions along with their Fourier coefficient amplitudes

For the computation purposes, one step of the forward transform can be expressed as the infinite matrix of wavelet coefficients represented below multiplied by the infinite signal vector.

$$
\begin{bmatrix} a_i \\ c_i \\ a_{i+1} \\ c_{i+1} \\ a_{i+2} \\ c_{i+2} \\ a_{i+3} \\ c_{i+3} \end{bmatrix}
=
\begin{bmatrix}
\dots & h_0 & h_1 & h_2 & h_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
\dots & g_0 & g_1 & g_2 & g_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \dots \\
\dots & 0 & 0 & h_0 & h_1 & h_2 & h_3 & 0 & 0 & 0 & 0 & 0 & \dots \\
\dots & 0 & 0 & g_0 & g_1 & g_2 & g_3 & 0 & 0 & 0 & 0 & 0 & \dots \\
\dots & 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & 0 & 0 & 0 & \dots \\
\dots & 0 & 0 & 0 & 0 & g_0 & g_1 & g_2 & g_3 & 0 & 0 & 0 & \dots \\
\dots & 0 & 0 & 0 & 0 & 0 & 0 & h_0 & h_1 & h_2 & h_3 & 0 & \dots \\
\dots & 0 & 0 & 0 & 0 & 0 & 0 & g_0 & g_1 & g_2 & g_3 & 0 & \dots
\end{bmatrix}
\bullet
\begin{bmatrix} s_i \\ s_{i+1} \\ s_{i+2} \\ s_{i+3} \\ s_{i+4} \\ s_{i+5} \\ s_{i+6} \\ s_{i+7} \end{bmatrix}
\tag{54}
$$

Each step of the wavelet transform applies both the scaling and wavelet function to the data input. The dot product (inner product) of the infinite vector and a row of the matrix produces either a smoother version of the signal a_i or a wavelet coefficient c_i. In an ordered wavelet transform, the smoothed a_i are stored in the in the lower half of the N element input vector while the wavelet coefficients c_i are stored in the upper half of the N element input vector. The algorithm is recursive and the smoothed values become the input to the next step.

However, using a standard dot product is grossly inefficient since most of the operands are zero. In practice the wavelet coefficient values are moved along the signal vector and a four element dot product is calculated. Since in equation (54) we assumed the infinite data set, we didn't have to worry about shifting the coefficients "off the end" of the signal. In fact we always deal with finite signal. Therefore if $i = N - 1$, the $i + 2$ and $i + 3$ elements will be beyond the end of the array. In such case the algorithm acts like the data is periodic, where the data at the start of the signal wraps around to the end with help of the modulo operation. Expressed in terms of arrays, for the forward transform this would be:

$$a[i] = h_0 s[i] + h_1 s[i+1] + h_2 s[(i+2) \bmod n/2] + h_3 s[(i+3) \bmod n/2]$$
$$c[i] = g_0 s[i] + g_1 s[i+1] + g_2 s[(i+2) \bmod n/2] + g_3 s[(i+4) \bmod n/2]$$

(55)

As a result of the D4 algorithm operation, the frequencies that were most prominent in the original ECG signal have appeared as high amplitudes in that region of the DWT signal that includes those particular frequencies. If the main information of the signal lies in the high frequencies, as happens most often, the time localization of these frequencies is more precise, since they are characterized by more number of samples. If the main information lies only at very low frequencies, the time localization is not very precise, since few samples are used to express signal at these frequencies. This procedure in effect offers a good time resolution at high frequencies, and good frequency resolution at low frequencies. Therefore the frequency bands with low amplitudes can be discarded without any major loss of information, allowing data reduction. An example of such ECG signal, normalized to unit amplitude, before and after the DWT transformation is illustrated in figure 7.14. Figure 7.14b shows the 3rd level Daubechies D4 wavelet transform of the original 128 samples long ECG signal in Figure 7.14a.
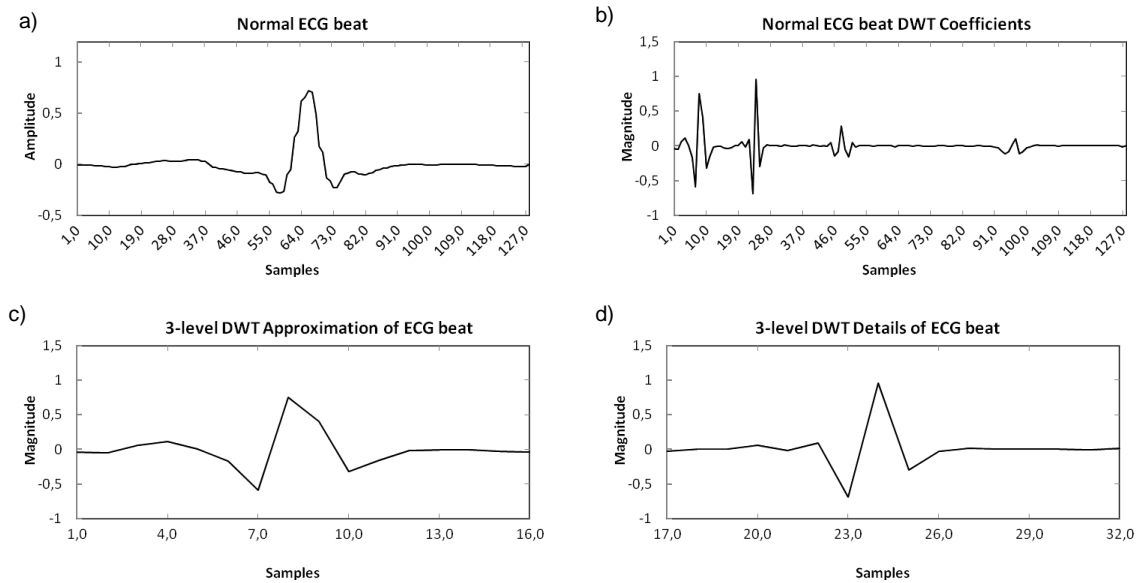
Figure 7.14 Example of a 3-level Daubechies D4 discrete wavelet transform performed on normal ECG beat.

The last 64 samples in this signal correspond to the highest frequency band in the signal, the previous 32 samples correspond to the second highest frequency band and so on. It should

be noted that only the first 32 samples, which correspond to lower frequencies of the analysis, carry relevant information and the rest of this signal has virtually no information. Therefore, all but the first 32 samples can be discarded without any loss of information,. This is how DWT provides a very effective data reduction scheme

The first 16 samples of the first 32 samples, illustrated in figure 7.14c, is the 3-level approximation of the original signal produced by the 3rd level low pass filter, which is the smother version of the original heartbeat. The second 16 samples, illustrated in figure 7.14d, are the 3-level DWT coefficients also called details of the ECG signal, containing signal characteristic in 12.5 Hz to 25 Hz frequency band, which is especially important for coarse grain heartbeat classification between sinus and ectopic wave. This is the classification according to the place of beat origin in the heart. Term "ectopic" simply mean that the heartbeat is coming from somewhere other than the sinus node in the heart. The difference between sinus and ectopic beats can be observed by comparing 3-level details of Normal and Ventricular ECG beats in Figure 7.14d and 7.15d respectively.
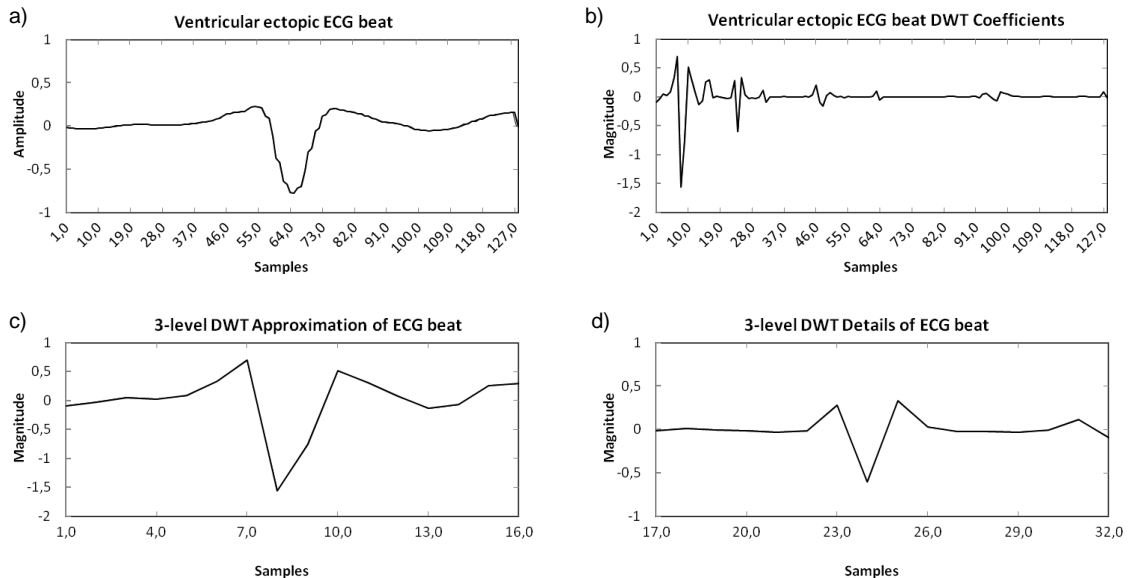
Figure 7.15 Example of a 3-level Daubechies D4 discrete wavelet transform performed on ventricular ectopic ECG beat.

7.6.1.2.2 Rhythm Analysis with RR interval ratio dispersion

The statistical analysis of the heartbeats distribution in the ECG signals, contained in the MIT-BIH Arrhythmia database, suggests that the instantaneous RR interval (interval between two successive R peaks) can potentially play an important role in the heart beat classification. It has been observed that, while beats belonging to class of Normal and/or Fusion beats (according to AAMI classification) maintain the RR interval ratio near or equal to 1, so for Supraventricular and Ventricular ectopic beats (premature beats) this ratio is rather less than 1. As long as this observation is mostly true for S and V beats which occur in separation, it is no longer true for S or V beats which occur in groups, where they local instantaneous RR interval is back again near or equal to 1. However for such cases the RR interval between current and the

previous beat will be very small compared to the mean RR of none premature beats in the ECG signal. For example, the difference in RR interval between Premature Ventricular Contraction (PVC), beat belonging to V class, and Atrial Premature (AP) beat , belonging to S class, is that a PVC is usually followed by a compensatory pause i.e., the RR interval between two QRS enclosing PVC beat equals twice the normal RR interval while the AP beat is usually followed by no compensatory pause i.e., the RR interval between two QRS enclosing AP beat is less than twice the normal RR interval (Ka 2012).

Based on these observations, a new statistic derived from the control charts theory, which analyse the rhythm of RR intervals, is considered in this study as the additional feature to the DWT compressed signal morphology features vector. It is the squared ratio of instantaneous RR interval to local moving average RR interval (calculated over the K number of RR intervals around the current RR interval), which satisfies the condition that the instantaneous RR is less than the two sample standard deviations s_N away from the local moving average RR interval. Otherwise the RR interval ratio is rounded up to 1. The equation (56) mathematically depicts this feature.

$$RR_{ratio}(k) = \begin{cases} \left(\dfrac{RR_k}{\overline{RR_k}}\right)^2 & if \quad RR_k < \overline{RR_k} - 2s_N \sqrt{\dfrac{1}{K-1}\sum_{i=0}^{N}(RR_i - \overline{RR_N})} \\ 1 & otherwise \end{cases} \qquad (56)$$

, where $\overline{RR_k} = \dfrac{1}{N}\sum_{i=k-K/2}^{k+K/2} RR_i$ is the local moving average of the RR interval,

and $s_N = \sqrt{\dfrac{1}{N-1}\sum_{i=0}^{N}(RR_i - \overline{RR_N})}$ is the standard deviation for the patient-specific

N representative normal RR intervals calculated for each patient individually. For this purposes typically first N number of normal heartbeats from within the first 5 minutes of each MIT-BIH record were used. According to AAMI recommended practice, such portion of the records can be used for training and algorithm adaptation purposes.

Figure 7.16 illustrates how the proposed RR_{ratio} can be used to classify heartbeats (for instance by using the Support Vector Machine (SVM)) to Normal and Abnormal according to their RR interval characteristics. Typical Abnormal heart beat, marked in red on the chart, characterise by the lower RR ratio when compared to Normal heartbeats, which are marked in blue on the following chart. By squaring these ratio the margin between the two classes maximize, what improves the separation of two classes in the classification features vector.
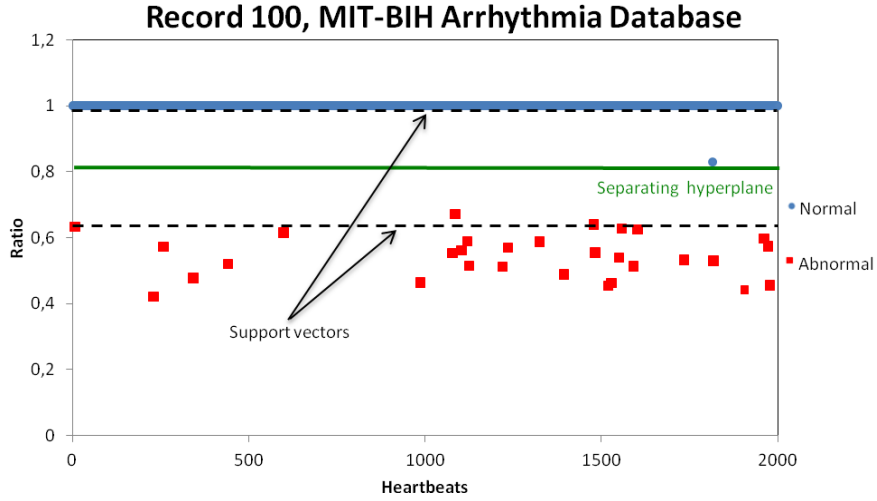
Figure 7.16 The graphical representation of RR interval ratio distribution for heartbeats in record 100 from the MIT-BIH arrhythmia database with hypothetical support vectors and separating hyperplane.

7.6.1.3 Support Vector Machine (SVM)

Support Vector Machines (SVMs) are primarily two-class classifiers with the distinct characteristic that aim to find the optimal hyperplane such that the expected generalization error (i.e., error for the unseen test patterns) is minimised. The optimization criterion is the width of the margin between the classes (i.e., the empty area around the decision boundary defined by the distance to the nearest training patterns) (Cristianini and Shawe-Taylor 2000).

The simplest case is the classification of objects belonging to two classes $y = \{-1; +1\}$ described by 2-dimensional input vectors $x \in R^2$ that are linearly separable, i.e., a linear classifier can perfectly separate them with a hyperplane. An illustrative example of such 2-class classification problem can be found in Figure 7.17. The aim of the training process is to find the maximum-margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. Primal version (Vapnik 1995) of the equation defining the decision function separating these classes is a hyperplane of the form:

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \mathbf{x} + b \tag{57}$$

, where $\mathbf{w}$ is a weight vector, $\mathbf{x}$ is the input vector, and b is the bias. The decision function (separating hyperplane) for the given classification problem can be obtained by selecting two hyperplanes H_1 and H_2 in a way that they separate the data and there are no points between them, and then try to maximise their distance. These hyperplanes can be described by inequalities (58)

$$H_1 : \mathbf{w}^T \cdot \mathbf{x}_i + b \geq +1 \;\; if \;\; y_i = +1$$
$$H_2 : \mathbf{w}^T \cdot \mathbf{x}_i + b \leq -1 \;\; if \;\; y_i = -1 \tag{58}$$

, what for separable case can be compacted to inequality (59) as follows.

$$y_i(\mathbf{w}^T \cdot \mathbf{x}_i + b) \geq 1 \;\; \forall i \tag{59}$$
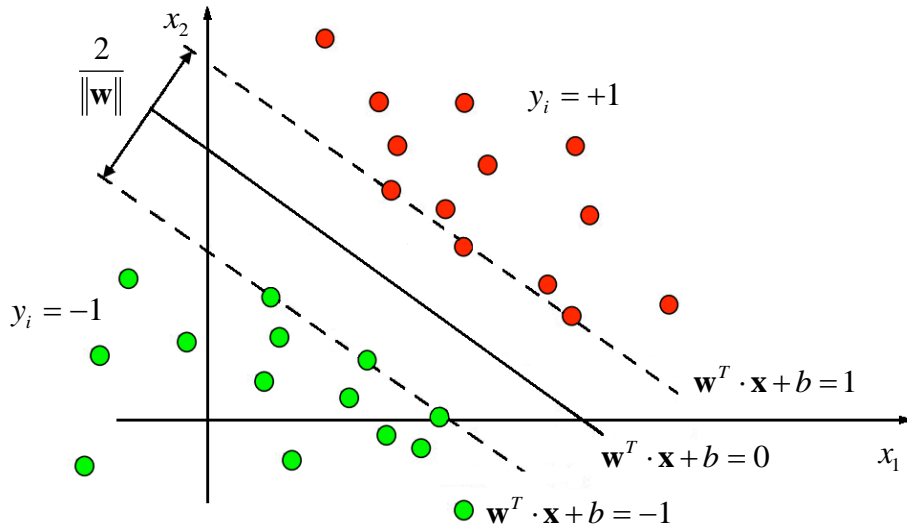
Figure 7.17 The optimal hyperplane separates positive and negative examples with the maximal margin.

The region bounded by H_1 and H_2 is called the margin and the hyperplane H_0 which is their median, is called the separating hyperplane. The margin can be calculated using the formula for the distance between a point and the line (Howard 1994). With distance between H_0 and H_1, the total distance between H_1 and H_2 is thus twice that as:

$$\frac{|\mathbf{w} \cdot \mathbf{x} + b|}{\|\mathbf{w}\|} = \frac{1}{\|\mathbf{w}\|} \quad \Rightarrow \quad \frac{2}{\|\mathbf{w}\|}$$

(60)

Therefore, in order to maximise the margin, we thus need to minimise $\|\mathbf{w}\|$. From computational point of view to minimise $\mathbf{w}$ in such form is difficult to solve because it depends on the norm of $\mathbf{w}$, which involves a square root. Fortunately for mathematical convenience it is possible to alter the equation by substituting $\|\mathbf{w}\|$ with $\frac{1}{2}\|\mathbf{w}\|^2$, without changing the solution as the minimum of the original and the modified equation have the same $\mathbf{w}$ and b.

Presented linearly separable case in practice is very rare, therefore Cortes and Vapnik (Cortes and Vapnik 1995) proposed a soft margin method that allows for mislabelled examples. They achieved it by introducing a non-negative slack (error) ξ_i variables, which measure the degree of misclassification. Such minimisation problem takes the primal form of:

$$\min_{w \in R^d, \xi_i \in R^+} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \right\}$$

(61)

subject to:

$$y_i(\mathbf{w}^T \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi \geq 0$$

The constraint can be written more concisely as $y_i f(\mathbf{x}_i) \geq 1 - \xi_i$, which, together with $\xi \geq 0$, is equivalent to:

$$\xi_i = \max(0, 1 - y_i f(\mathbf{x}_i))$$

(62)

Hence the learning problem is equivalent to the primal unconstrained optimisation problem over $\mathbf{w}$ in form of:

$$\min_{w \in R^d} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i}^{n} \max(0, 1 - y_i f(\mathbf{x}_i)) \right\} \tag{63}$$

$$\underbrace{\qquad\qquad}_{\text{Maximise margin}} \underbrace{\qquad\qquad\qquad\qquad}_{\text{Minimise training terror}}$$

The result is a hyperplane that minimises the sum of errors ξ_i while maximising the margin for the correctly classified data. For the introduced loss function, points are in two categories:

1) $y_i f(\mathbf{x}_i) \geq 1$ - point is outside/on margin therefore no contribution to loss.

2) $y_i f(\mathbf{x}_i) < 1$ - point violates margin constraint therefore contributes to loss.

The constant C controls the trade-off between margin and misclassification errors what aims to prevent outliers from affecting the optimal hyperplane. Large C (narrow margin) decrease the training error but increase the risk of poor generalization.

7.6.1.3.1   SVM training with Structural Risk Minimization (SRM)

Instead of directly minimising the empirical risk calculated from the training data, SVMs perform Structural Risk Minimisation to achieve good generalization on a limited number of learning patterns (Vapnik 1999). Unlike Empirical Risk Minimisation used for example in Artificial Neural Networks, the SRM uses a set of models ordered in terms of their complexities (an example is polynomials of increasing order), where model selection corresponds to finding the model simplest in terms of order and best in terms of empirical error on the data. The procedure is outlined below.

1. Using a priori knowledge of the domain, choose a class of functions, such as polynomials of degree n.
2. Divide the class of functions into a hierarchy of nested subsets in order of increasing complexity. For example, polynomials of increasing degree.
3. Perform conventional Empirical Risk Minimisation on each subset (this is essentially parameter selection).
4. Select the model in the series whose sum of empirical risk and order are minimal.

Conventional classification problem attempts to learn the mapping function given as:,

$$y = f(\mathbf{x}, \alpha) \tag{64}$$

where $x \in X$ is n-dimensional vector representing some object, $y \in Y$ is a class label and α are the parameters of the function. As such the generalisation should be obtained, where given a previously seen $x \in X$ it finds a suitable $y \in Y$.

With the conventional Empirical Risk Minimisation (Vapnik 1995), the generalization is obtained by choosing a function that perform best on the training data, according to empirical

risk also called the training error. Given a function f and a training set S consisting of n data points, we can measure the empirical risk of f, represented by its coefficients vector α, as:

$$R_{emp}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} \ell\big(f(\mathbf{x}_i, \alpha), y_i\big)$$

(65)

, where ℓ is the zero-one loss function $\ell(y, \hat{y}) = 1$ if $y \neq \hat{y}$, and 0 otherwise.

The drawback of the conventional Empirical Risk minimisation over training data is, that it does not imply good generalisation to novel test data. Firstly, there could be a number of different functions, which all give a good approximation to the training data set, but not training set, and secondly it is difficult to determine a function which best captures the true underlying structure of the data distribution when aiming to minimise the overall risk, called Test Error (Vapnik 1995), given by equation (66):

$$R(\alpha) = \int \ell\big(f(\mathbf{x}, \alpha), y\big) dP(\mathbf{x}, y)$$

(66)

where $P(\mathbf{x}, y)$ is the unknown joint distribution of $\mathbf{x}$ and y.

Based on the training data alone, there is no means of choosing which function is better. In this case a generalisation is not guaranteed therefore a restriction must be placed on the allowed functions. To guarantee an upper bound on generalisation error, statistical learning theory says that the capacity of the learned functions must be controlled. Structural Risk Minimisation aims to address this problem and provides a well-defined quantitative measure of the capacity of a learned function to generalize over unknown test data. To control the generalization ability of a learning machine one has to control two different factors: the empirical risk and the capacity of the learning machine as measured by its VC dimension (Cherkassky, Xuhui, Mulier *et al.* 1999). Therefore the upper bound on the true risk can be given by inequality

$$R(\alpha) = R_{emp}(\alpha) + \sqrt{\frac{h\big(\log\big(\frac{2n}{h}\big)+1\big) - \log\big(\frac{\eta}{4}\big)}{n}}$$

(67)

, where n is the length of the training multi-set, η is the confidence, where $0 \leq \eta \leq 1$ and h is the VC dimension of the set of functions parameterised by α. The VC dimension of a set of functions is the maximum number of points that can be separated in all possible ways by that set of functions. For linear classifiers the VC dimension for a hyperplane in R^n, can be shown to be $n+1$. Note that the above upper bound is independent of the true distribution.

The equation (75) can be further simplified to:

Test Error $\leq$ Training Error + Complexity of set of Models

Therefore the training process is a trade-off between the quality of the approximation of the given data and the complexity of the approximating function. As illustrated in Figure 7.18 if high capacity set of functions is taken then low training error can be obtained but the risk of over fitting the model is higher. On the other side with a very simple set of models of low complexity, the low training error won't be obtained, what will result in under fit model.
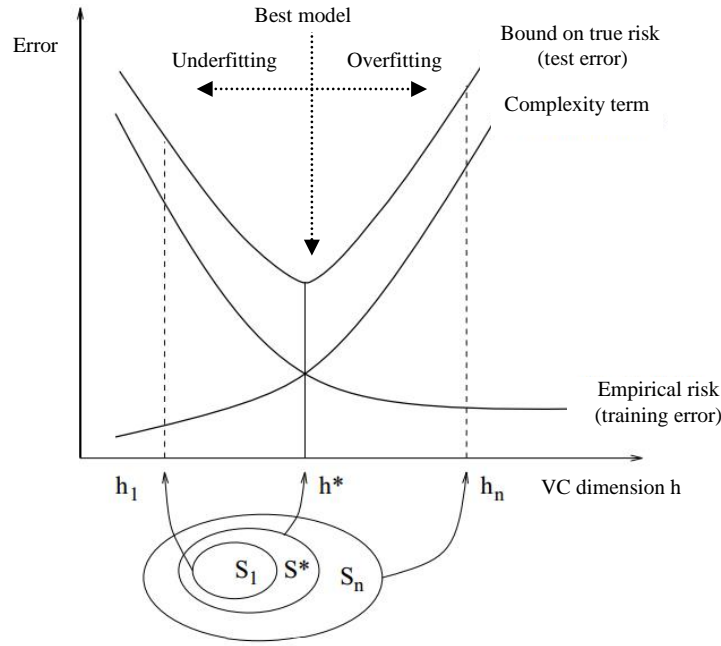
Figure 7.18 Schematic illustration of trade-off between the quality of the approximation of the given data and the complexity of the approximating function.

7.6.1.3.2   Kernel Trick

In practice, however, linear classifiers aren't complex enough to deal with most of the problems. The Support Vector Machine solution to this are nonlinear SVMs, which states that any learning problem can be made easy with the right set of features being taken into account. The idea behind the nonlinear SVMs is to make use of a nonlinear mapping function Φ that transforms data in input space to data in feature space in such a way as to render a problem linearly separable. The SVM then automatically discovers the optimal separating hyper-plane (OSH), which maximizes the margin between the two nearest data points belonging to two separate classes, The OSH when mapped back into input space via Φ^{-1}, can be a complex decision surface.

Formally, the nonlinear mapping function pre-process the data with:

$$x \rightarrow \Phi(x) \tag{68}$$

and then learn the decision function of primal form as:

$$f(\mathbf{x}) = \mathbf{w}^T \cdot \Phi(\mathbf{x}) + b \tag{69}$$

It is often the case that in order to effectively separate the data, we must use a feature space that of higher dimension than the input space. Therefore in formulation of Φ is applicable the kernel trick, which improves the capacity of these algorithms. In this case learning is not being directly performed in the original space of data but in a new space called feature space. For this reason SVM is the representative of so called Kernel Machines (KMs).

Note: feature space is of higher dimension
than the input space in practice
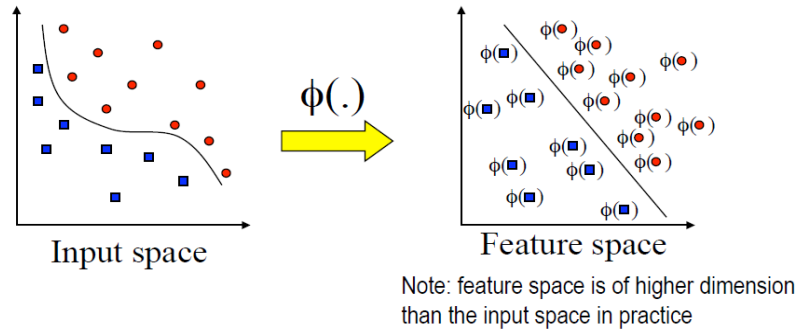
Figure 7.19 The kernel function calculates inner products in the future space.

The problem in such case is however the dimensionality of $\Phi(x)$ which can be very large, making $\mathbf{w}$ hard to solve and represent explicitly in memory. Thanks to Representer Theorem (Kimeldorf and Wahba 1970) which states that the solution $\mathbf{w}$ can always be written as a linear combination of the training data $\mathbf{x}_i$ (and therefore also mapped to future space with $\Phi(x)$) as:

$$w = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x_i} \quad \overset{\Phi(.)}{\rightarrow} \quad \mathbf{w} = \sum_{i=1}^{N} \alpha_i y_i \Phi(\mathbf{x}_i) \tag{70}$$

Therefore, the SVM decision function from equation (57) can be reformulated to learn a non-linear classifier (Platt 1999):

$$f(\mathbf{x}) = \sum_i \alpha_i y_i \Phi(\mathbf{x}_i)^T \Phi(\mathbf{x}) + b = \sum_i \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}) + b \tag{71}$$

by solving an optimization problem over Lagrange multipliers $\alpha_i \ i = 1,...,N$ one for each of the inequality (59) constraints (Burges 1998), instead of optimizing $\mathbf{w}$ directly what would require to compute $\Phi(x)$ explicitly. The primal unconstrained optimization problem over $\mathbf{w}$ from equation (63) by substituting $\|\mathbf{w}\|^2$:

$$\|\mathbf{w}\|^2 = \left\{\sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i\right\}^T \left\{\sum_{j=1}^{N} \alpha_j y_j \mathbf{x}_j\right\} = \frac{1}{2}\sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j (\mathbf{x_i}^T \mathbf{x_j}) \tag{72}$$

can now be formulated in dual form over α_i as:

$$\max_{\alpha_i \geq 0}\left\{\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j(\mathbf{x_i}^T\mathbf{x_j})\right\} \overset{\Phi(.)}{\rightarrow} \max_{\alpha_i \geq 0}\left\{\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j K(\mathbf{x_i},\mathbf{x_j})\right\} \tag{73}$$

subject to the inequality constraints:

$$0 \leq \alpha_i \leq C \ \ for \ \ \forall i \tag{74}$$

and one linear equality constraint:

$$\sum_{i=1}^{N} \alpha_i y_i = 0 \tag{75}$$

Such formulated optimization problem can be solved by quadratic programming (QP) optimization, which in more details is described by (Floudas and Visweswaran 1995).

The $K(\cdot,\cdot)$ in the above equations is called the kernel function. Depending on the chosen Φ, the future space dimension might possibly be high or even infinite, so working with Φ directly and calculating dot products of mapped points would be difficult. Kernel function enables the SVM to operate in a high-dimensional, implicit feature space without ever computing the coordinates of the data in that space, but rather by simply computing the inner products between the images of all pairs of data in the feature space. To illustrate how kernel functions overcome this problem, let's consider the two points $\mathbf{x}=(x_1,x_2)$, $\mathbf{x}'=(x_1',x_2')$ and apply the mapping Φ, which increase the dimension of the vector on x and x' as:

$$\Phi:R^2\to R^3\quad(x_1,x_2)\to(z_1,z_2,z_3):=(x_1^2,\sqrt{2}x_1x_2,x_2^2)$$

$$\begin{aligned}\Phi(x_1,x_2)\cdot\Phi(x_1',x_2')&=(x_1^2,\sqrt{2}x_1x_2,x_2^2)\cdot(x_1'^2,\sqrt{2}x_1'x_2',x_2'^2)\\&=x_1^2x_1'^2+2x_1x_1'x_2x_2'+x_2^2x_2'^2\\&=(x_1x_1'+x_2x_2')^2\\&=((x_1,x_2)\cdot(x_1',x_2'))^2=(\mathbf{x}\cdot\mathbf{x}')^2=:K(\mathbf{x},\mathbf{x}')\end{aligned}\tag{76}$$

At the end we can see that instead of calculating this particular mapping followed by the dot product, we can equivalently calculate the dot product of the original points and square it. Thus we can calculate the dot product of $\Phi(\mathbf{x})\cdot\Phi(\mathbf{x}')$ without applying the function Φ. Such functions that are equivalent to mapping followed by a dot product in called kernel functions. Kernel function K is used as a similarity measure for $\mathbf{x}$ and $\mathbf{x}'$ without explicitly knowing Φ nor the dimension of the future space.

An "ideal" kernel function assigns a higher similarity score to any pair of objects that belong to the same class than it does to any pair of objects from different classes. This is the case if the implicit mapping by the kernel function brings similar objects close together and takes dissimilar objects apart from each other in the induced feature space. Table 7.8 lists some of the most frequently used kernel functions (Genton 2002, Vert, Tsuda and Schölkopf 2004).

Name of Kernel	Kernel Function*
Linear kernel	$K(\mathbf{x},\mathbf{x}')=\mathbf{x}^{\mathbf{T}}\cdot\mathbf{x}'+c$
Polynomial kernel	$K(\mathbf{x},\mathbf{x}')=(\alpha\mathbf{x}^{\mathbf{T}}\cdot\mathbf{x}'+c)^d$
Gaussian Radial basis function (RBF) kernel	$K(\mathbf{x},\mathbf{x}')=\exp\left(-\dfrac{\lVert\mathbf{x}-\mathbf{x}'\rVert^2}{2\sigma^2}\right)$
Sigmoid kernel	$K(\mathbf{x},\mathbf{x}')=\tanh(\alpha x^T\cdot x'+c)$

*) α, c, d are kernel-specific parameters, and $\sigma^2=mean\lVert x-x'\rVert^2$.

Table 7.11 Most frequently used kernel functions

7.6.1.4   Cardiac arrhythmia detection procedure

The proposed cardiac arrhythmia detection procedure, consists of three main stages: 1) a pre-processing stage, 2) features extraction stage and 3) classification stage. Figure 7.20 graphically depicts all the stages of a prototype system, based on the outcomes of the above study of Wavelet Transforms and Support Vector Machines.
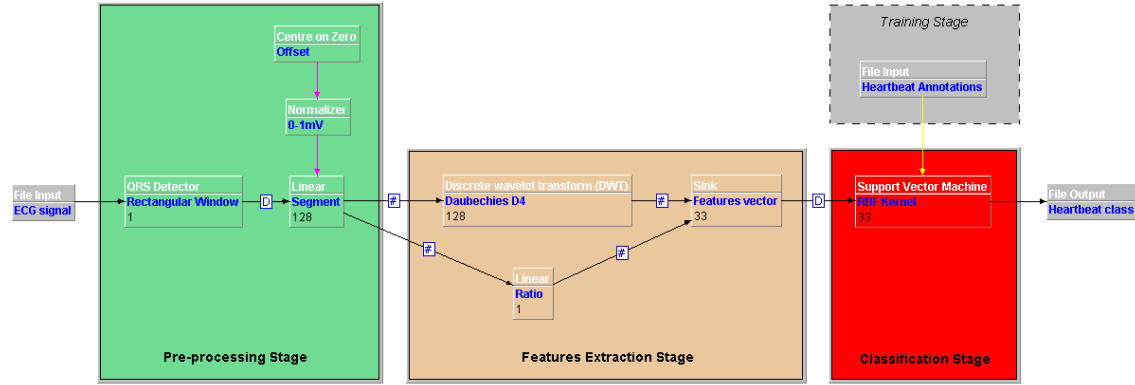
Figure 7.20 Block diagram of cardiac arrhythmia detection procedure in jMENN Editor.

In the presented model the digitized ECG signal is applied at the input to the pre-processing stage. Since the ECG waveforms of different patients and patient groups vary widely in their morphologies that affect feature extraction, the pre-processing stage focus on heart beat segmentation and normalization. The segmentation is obtained with rectangular window, centred on the R-peak, covering QRS onset and offset, each of ±270ms (≈±100 samples) long, which is further down sampled to 128 samples long segment. The QRS detection algorithm used to obtain the location of the fiducial points (R-peaks) in the continuous ECG signal is presented and discussed in details in section 7.5.1 of this chapter. Such obtained segments are then subject to normalization which ensure that the classification does not depend on the maximum amplitude of the ECG record nor its offset. Thus the peak-to-peak magnitude of each segment is normalized to a value of 1mV followed by the mean value of that segment being fixed to zero value.

Pre-processed segments of ECG signal form the input to the features extraction stage, which is concerned with producing a vector of measurements, called features vector for each heartbeat. Such features vector is the then processed by the classifier. Classifiers using the features extraction module, which discriminate smaller number of characteristics features, achieve greater classification performance, when compared to the same classifiers using direct ECG samples. However the choice of the right set of those features is crucial. The proposed algorithm use a combination of morphological and statistical features, which deliver meaningful information on both the shape of the heart beat and its rhythm. What determined such set of features is the traditional manual ECG signal interpretation, where both information are needed to explicitly assign the heart beat to a normal or abnormal class. The first part of the features extraction focus on the shape of the heartbeat. The proposed morphological features extraction method involves the presented in Section 7.6.1.2.1 Daubechies D4 discrete wavelet transform

(DWT) of a segment of length 128 around the R-peak. The main advantage of using Daubechies wavelet is their similarity in shape to QRS complex with their energy spectrum concentrated around low frequencies. It allows to eliminate any remaining noise in the segment by extract signal details in a way that an accurate approximation of main signal components of reduce dimension can be produced. To determine what is the best configuration of DWT part of the features vector, four different wavelet coefficients were evaluated during the validation process. This included the 2nd level approximation (32 samples), 3rd level approximation and details (32 samples), 3rd level approximation only (16 samples), and 3rd level details only (16 samples). Complete results are presented in section 7.6.1.6. The second part of the features extraction focus on the heartbeat rhythm analysis using the RR interval ratio dispersion presented in section 7.6.1.2.2. Outputs from this two features extraction methods are combined into one features vector by the sink layer which further provide the input to the classifier.

For the classification of the cardiac beats the Support Vector Machine (SVM), in details presented in Section 7.6.1.3 is used. As a supervised machine learning method, SVM requires a set of well annotated training data to learn how to distinguish the normal beat from the abnormal one. Therefore, as illustrated in the block diagram in Figure 7.20, during the training process an additional file input with target data must be presented to the classifier. The training phase consists of finding a set of support vectors and their parameters to determine the decision boundary between two classes, by solving a Quadratic Programming (QP) optimization problem, for which many effective techniques have been proposed (Keerthi, Shevade, Bhattacharyya *et al.* 2001, Serafini, Zanghirati and Zanni 2005, Bottou and Lin 2007)

In this study a modified version of QP, called Sequential Minimal Optimization (SMO) presented in detail by Platt (Platt 1999) is used. SMO is a simple algorithm that quickly solves the SVM QP problem without any extra matrix storage and without invoking an iterative numerical routine for each sub-problem. SMO decomposes the overall QP problem into QP sub-problems and chooses to solve the smallest possible optimization problem at every step analytically. Considering the standard SVM QP problem defined in equation (81), the smallest possible optimization problem involves two Lagrange multipliers α_i because of the linear equality constraint (83). At every step, SMO chooses two Lagrange multipliers to jointly optimize, finds the optimal values for these multipliers, and updates the SVM to reflect the new optimal value. In order to solve for the two Lagrange multipliers, SMO first computes the constraints on these multipliers and then solves for the constrained minimum. For any two of such α_1 and α_2, the constraints (82) and (83) can be easily displayed in two dimensions as in Figure 7.21. The bound constraints (82) cause the Lagrange multipliers to lie within a box, while the linear equality constraint (83) causes the Lagrange multipliers to lie on a diagonal line. The two Lagrange multipliers must fulfil all of the constraints of the full problem. Thus, the constrained minimum of the objective function must lie on a diagonal line segment.

Considering the QP problem in equation (73) the problem is solved when for all i the Lagrange multipliers satisfy the Karush-Kuhn-Tucker (KKT) conditions (Platt 1999) (within a user-defined tolerance), as follows:

$$\alpha_1 = 0 \Leftrightarrow y_i u_i \geq 1$$
$$0 < \alpha_i < C \Leftrightarrow y_i u_i = 1 \tag{77}$$
$$\alpha_i = C \Leftrightarrow y_i u_i \leq 1$$

where u_i is the output of the SVM and y_i is the target for the i th training example. The KKT conditions can be evaluated on one example at a time, what is used by the SMO algorithm. Without loss of generality, we can optimise two multipliers α_1^{new} and α_2^{new} at a time, from an old set of feasible solution α_1 and a_2. Because of constrained (75), we have:

$$\alpha_1 y_1 + \alpha_2 y_2 + \sum_{i=3}^{N} \alpha_i y_i = 0 \tag{78}$$

we can formulate it as:

$$\alpha_1 y_1 + \alpha_2 y_2 = k, \quad where \quad k = -\sum_{i=3}^{N} \alpha_i y_i \tag{79}$$

Without loss of generality, the algorithm first computes the second Lagrange multiplier α_2 and computes the ends of the diagonal line segment in terms of α_2. Bearing in mind that $y_i \in \{-1,1\}$ there are two cases to consider when evaluating two training samples, as illustrated in Figure 7.21. The first case is when the target y_1 does not equal the target y_2, then the ends of the diagonal line segment in terms of α_2 can be expressed as follows:

L (α_2 at the lower end point) is: $\max(0,-k) = \max(0,\alpha_2 - \alpha_1)$

H (α_2 at the higher end point) is: $\min(C,C-k) = \min(C,C+\alpha_2 - \alpha_1)$
$$\tag{80}$$

The second case is when the target y_1 equals the target y_2, then the following bounds apply to α_2:

L (α_2 at the lower end point) is: $\max(0,k-C) = \max(0,\alpha_1 + \alpha_2 - C)$

H (α_2 at the higher end point) is: $\min(C,k) = \min(C,\alpha_1 + \alpha_2)$
$$\tag{81}$$
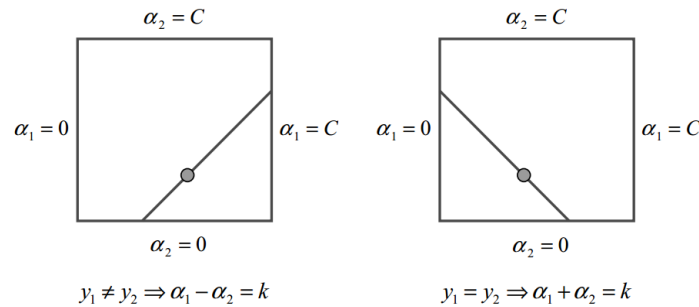
Figure 7.21 Two cases of optimizations (Platt 1999).

Fixing the other α_i's of the objective function (73) and solving it for the α_2 by taking the first and second derivatives of this function, will find the minimum along the direction of the linear equality constraint. The second derivative η, used to test for local extrema of the objective function along the diagonal line, can be expressed as :

$$\eta = K(\mathbf{x_1}, \mathbf{x_1}) + K(\mathbf{x_2}, \mathbf{x_2}) - 2K(\mathbf{x_1}, \mathbf{x_2}) \tag{82}$$

Under normal circumstances, the objective function will have a minimum along the direction of the linear equality constraint, when η (second derivative) will be greater than zero. In this case, SMO computes the minimum along the constraint:

$$\alpha_2^{new} = \alpha_2 + \frac{y_2(E_1 - E_2)}{\eta} \tag{83}$$

where $E_i = u_i - y_i$ is the error on the i th training example. As a next step, the constrained minimum is found by clipping the α_2^{new} minimum to the ends of the line segment:

$$\alpha_2^{new,clipped} = \begin{cases} H & \text{if } \alpha_2^{new} \geq H \\ \alpha_2^{new} & \text{if } L < \alpha_2^{new} < H \\ L & \text{if } \alpha_2^{new} \leq L \end{cases} \tag{84}$$

Let $s = y_1 y_2$ and multiply the equation (79) by y_1 and we will get:

$$\alpha_1^{new} = k - s\alpha_2^{new} \tag{85}$$

where $k = \alpha_1^{new} + s\alpha_2^{new} = \alpha_1 + s\alpha_2$. Thus the value of α_1^{new} can be computed from the new, clipped α_2^{new} as:

$$\alpha_1^{new} = \alpha_1 + s(\alpha_2 - \alpha_2^{new,clipped}) \tag{86}$$

We might find that $\eta = 0$ when there is more than one example in our training set having the same input vector $\mathbf{x}$. In this case the algorithm will find another training example and re-optimize the first multiplier with a new α_2. In unusual circumstances, η will not be positive. In such case the SMO evaluates the objective function Ψ at each end of the line segment according to the following equations according to (Platt 1999):

$$f_1 = y_1(E_1 + b) - \alpha_1 K(\mathbf{x_1}, \mathbf{x_1}) - s\alpha_2 K(\mathbf{x_1}, \mathbf{x_2})$$

$$f_2 = y_2(E_2 + b) - \alpha_2 K(\mathbf{x_2}, \mathbf{x_2}) - s\alpha_1 K(\mathbf{x_1}, \mathbf{x_2})$$

$$L_1 = \alpha_1 + s(\alpha_2 - L)$$

$$H_1 = \alpha_1 + s(\alpha_2 - H) \tag{87}$$

$$\Psi_L = L_1 f_1 + L f_2 + \tfrac{1}{2} L_1^2 K(\mathbf{x_1}, \mathbf{x_1}) + \tfrac{1}{2} L^2 K(\mathbf{x_2}, \mathbf{x_2}) + sLL_1 K(\mathbf{x_1}, \mathbf{x_2})$$

$$\Psi_H = H_1 f_1 + H f_2 + \tfrac{1}{2} H_1^2 K(\mathbf{x_1}, \mathbf{x_1}) + \tfrac{1}{2} H^2 K(\mathbf{x_2}, \mathbf{x_2}) + sHH_1 K(\mathbf{x_1}, \mathbf{x_2})$$

SMO will move the Lagrange multipliers to the end point that has the lowest value of the objective function. If the objective function at both ends is the same within a small round-off error of ε, then the joint optimization makes no progress. These two α are skipped and another pair of multipliers to optimize is selected.

Summarising the algorithm proceeds as follows:

1. It finds a Lagrange multiplier α_1 that violates the Karush–Kuhn–Tucker (KKT) conditions for the optimization problem.

2. It picks a second multiplier α_2 and optimize the pair (α_1, α_2).

3. It repeat steps 1 and 2 until convergence.

The detailed pseudo-code of the SMO algorithm can be found in (Platt 1999).

7.6.1.5 Model selection

The effectiveness of SVM does not only depend on the selection of kernel but also the kernel's parameters and soft margin parameter C, which are in general referred to as hyperparameters. In search for the optimal α coefficients, a set of such hyperparameters, must be tuned in order to obtain good generalization This phase is called model selection and is strictly linked with the estimation of the generalization ability of a classifier (i.e., the error rate attainable on new and previously unobserved data), as the chosen model is characterized by the smallest estimated generalization error. Unfortunately, the tuning of the hyperparameters is not a trivial task (Anguita, Ghio, Greco *et al.* 2010) as there are usually multiple parameters to tune at the same time and moreover, the estimates of the error are not explicit functions of these parameters. In practice there are only four common kernels mentioned in Table 7.8, therefore the model selection typically starts from the decisions which one to try first, followed by the penalty parameter C and corresponding kernel parameters optimization.

In general, the Gaussian Radial Basis Function (RBF) kernel is a reasonable first choice. This kernel nonlinearly maps samples into a higher dimensional space so it, unlike the linear kernel, can handle the case when the relation between class labels and attributes is nonlinear. It does it by adding a "bump" around each data point. As shown by Keerthi and Lin (2003) the linear kernel is a special case of RBF since the linear kernel with a penalty parameter C has the same performance as the RBF kernel with some parameters (C, γ). Furthermore, Lin and Lin (Lin and Lin 2003) showed that sigmoid kernel behaves like RBF for certain parameters and in general is not better than the RBF kernel. Therefore if complete model selection using the Gaussian kernel has been conducted, there is no need to consider linear nor sigmoid SVM. The second reason behind use of RBF is the number of hyperparameters, which influence the complexity of model selection. In contrast to polynomial kernels of which kernel values may go to infinity, the RBF kernel takes only one kernel parameter γ (in addition to SVM soft margin parameter C). Since it is not known beforehand which C and γ are best for a given problem consequently the model selection must be done.

Therefore the goal of model selection is to find good (C, γ) so that the classifier can accurately predict unknown data (i.e. test data). The prediction accuracy obtained from the "unknown" set more precisely reflects the performance on classifying an independent data set. A common strategy is to separate the training data set into two parts, of which one is considered unknown, and using a grid-search method (Hsu, Chang and Lin 2010) on C and γ parameters, to choose the model with best validation accuracy as the best choice for the problem. The grid-search may seem naive but it is a straightforward method which tries various pairs of (C, γ) values and select the one with the best cross-validation accuracy. Compared with several advanced methods (Anguita, Boni, Ridella *et al.* 2005, Wu and Wang 2009) which can save computational cost by, for example, approximating the cross-validation rate or using heuristics in parameter selection, the grid search characterise by an exhaustive parameter search, capable to avoid local extrema and deliver more accurate and confident results. Moreover, unlike many of the advanced methods which are iterative processes, the grid-search can be easily parallelized because each (C, γ) pair is independent. Therefore, the computational time required to find good parameters by grid-search is not much more than that by advanced methods. However in order to further speed-up the model selection a coarse grid search is conducted first, followed by the finer grid search on the region with highest accuracy (Hsu, Chang and Lin 2010).

The model validation technique, used to assess how the results of a statistical analysis will generalize to an independent data, adopted in this research is known as k-fold cross-validation. In k-fold cross-validation procedure, the original sample is randomly partitioned into k equal size subsamples of which a single subsample is retained as the validation data for testing the model, and the remaining $k-1$ subsamples are used as training data. This is then repeated k times, what ensures that all observations are used for both training and validation, and each observation is used for validation exactly once. In general k remains an unfixed parameter but the greater its value is, the more exhaustive, complex, and time consuming the validation of one model is. For this reason the 2-fold cross validation scheme was selected what ensures that the training and test sets are both large.

7.6.1.6 Experimental Results

The DS1 training data set for ECG arrhythmias used in this study was taken from MIT–BIH Arrhythmias Database (Moody and Mark 2001). All training patterns sampled at 360 Hz, were first arranged as 200 samples long R–R intervals, referred to as a segments. Each segment covered 100 samples to the left and 100 samples to the right of the R peak. Each of these segments was further down-sampled to 128 samples, for which a ratio and the 3rd order Discrete Wavelet Transform (DWT) was obtained. This resulted in 33 samples long segments for each ECG beat. Due to a large number of those segments in the DS1, only 6000 of them

were randomly selected and used to form two equal size validation sub sets, each containing 1500 normal and 1500 abnormal ECG beats. In order to ensure that morphological differences between subjects as well as all types of ECG abnormalities are well represented in both validation subsets, 7% of all normal beats and 62% of all abnormal beats were randomly selected from each record of the DS1 data set. These is a result of much larger number of Normal beats than Abnormal beats in MIT-BIH Arrhythmia Database.

In order to find the optimum values for the penalty parameter C and kernel parameter γ on such validation dataset, the grid-search method was used. For each pair of C and γ values, three statistical measures of a classification performance, for each test run, were observed according to (American National Standards Institute 1998, Kohler, Hennig and Orglmeister 2002). These are:

1) Accuracy (Acc) – which measures the degree of closeness of classification to actual true value, defined as:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

(88)

2) Sensitivity (Se) – measures the proportion of actual positives which are correctly identified as such (e.g. the percentage of abnormal beats which are correctly identified), defined as

$$SE = \frac{TP}{TP + FN} \times 100\%$$

(89)

3) Specificity (Sp) – measures the proportion of negatives which are correctly identified (e.g. the percentage of normal beats which are correctly identified) defined as:

$$SP = \frac{TN}{FP + TN} \times 100\%$$

(90)

These measures are computed based on confusion matrix. Confusion matrix is a specific table that allows visualization of the performance of an algorithm. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. It makes easy to see if the system is confusing two classes (i.e. commonly mislabelling one as another). The following table is the sample confusion matrix.

		Condition (as determined by a doctor)	
		Abnormal	Normal
Test Outcome	Abnormal	True Positive	False Positive (Type I error)
	Normal	False Negative (Type II error)	True Negative

Figure 7.22 Confusion matrix.

Pertaining to a specific class, each cell in such matrix can be labelled as true positive (TP), false positive (FP), true negative (TN) and false negative (FN), where:

- true-positive (TP), (equivalent with hit): the number of heart beats, which have been correctly assigned to a certain class (abnormal to abnormal);
- false-positive (FP) (equivalent with false alarm, Type I error): the number of heart beats, that actually belong to other classes but were incorrectly assigned to this specific class (normal to abnormal);
- true-negative (TN) (equivalent with correct rejection): the number of heart beats, which have been correctly assigned to other class (normal to normal);
- false-negative (FN) (equivalent with miss, Type II error): the number of beats which actually belong to that class, but were misclassified and were assigned to other classes (abnormal to normal).

For such defined performance measurements the grid search was performed following Hsu et al. (Hsu, Chang and Lin 2010) on exponentially growing sequences of $C = 2^{-5}, 2^{-3}, \ldots, 2^{5}$ and $\gamma = 2^{-5}, 2^{-3}, \ldots, 2^{5}$ parameters. In this way three grids in the (C, γ) parameter space were obtained, which are presented in Figure 7.23, 24 and 25.
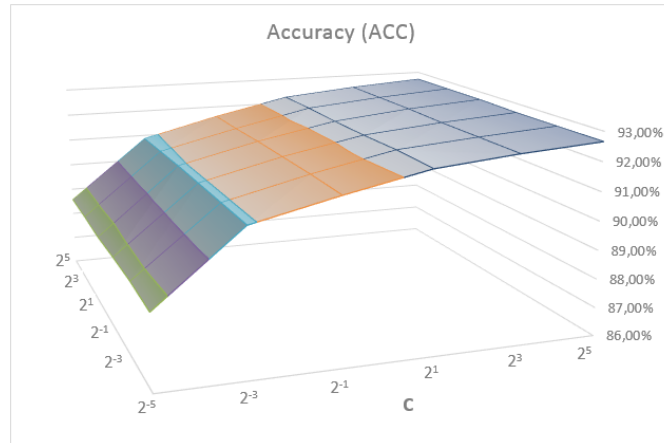
Figure 7.23 Accuracy grid.
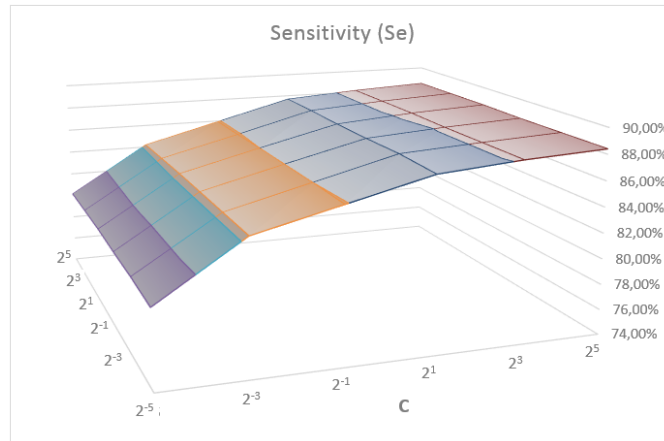
Figure 7.24 Sensitivity grid.
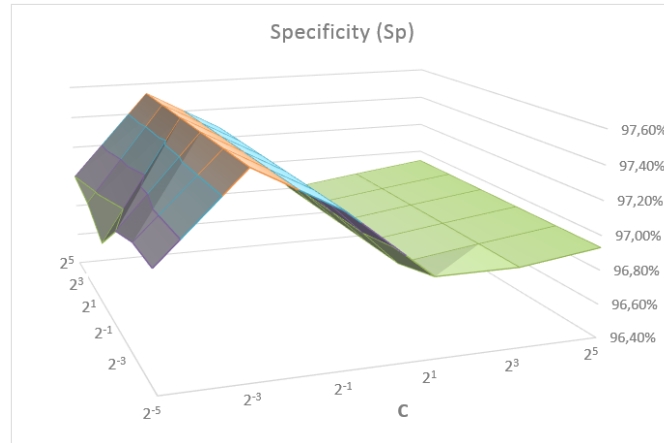
Figure 7.25 Specificity grid.

Results show that the performance of the SVM algorithm on the given classification problem significantly depends on the penalty parameter C, while the kernel parameter γ is insignificant. By combining all three grids into one single grid search, presented in Figure 7.26, and contrasting this with the grid showing average number of SMO algorithm iterations needed to train the model, presented in Figure 7.27, it was found out that that the most optimal pair of (C, γ), which produce best results after relatively short number of SMO iterations is $(2^3, 2^1)$.
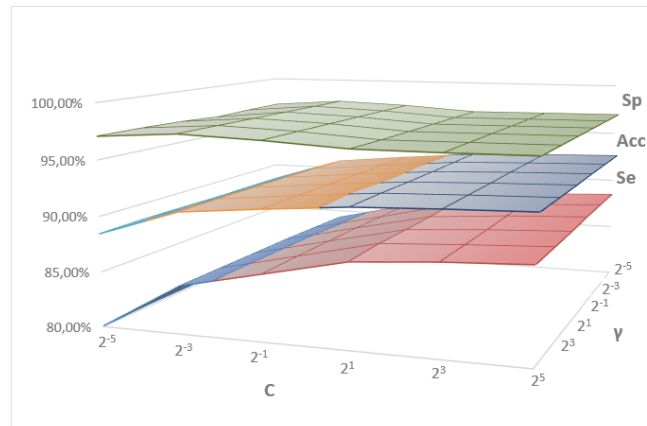
Figure 7.26 Combined Accuracy, Sensitivity and Specificity grids.
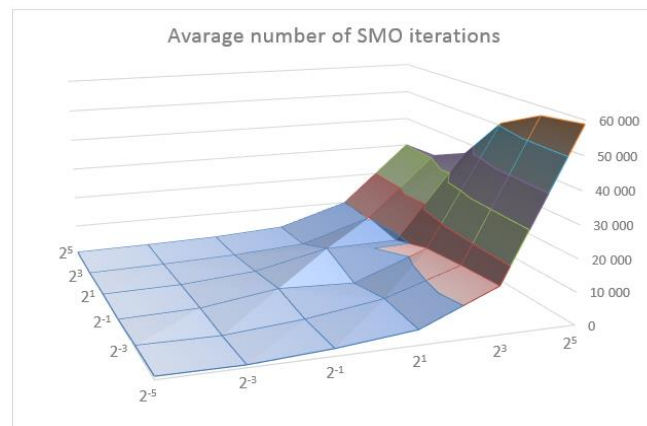
Figure 7.27 Average number of Sequential Minimal Optimisation (SMO) algorithm iterations.

After the best (C, γ) was found, the SVM classifier was once again retrained using the whole DS1 and tested on DS2 data set. Such assessment is unbiased as DS2 was not used at any point in the development of the classifier. In accordance with AAMI recommendations, the classification performance for each recording as well as statistical performance figures were calculated for DS2 (test-set) and are shown in Table 7.12. The best performing classifier produced in total 825 false negatives (FN), and 695 false positives (FP) out of total of 49 672 heart beats. This resulted in the overall detection accuracy (Acc) of 96.94%, sensitivity (Se) at the level of 84.84%, and specificity (Sp) at the level of 98.43%.

Looking at individual ECG records the detection accuracy vary from 88.12% to 99.91%, sensitivity vary from 37.93% to 100%, and specificity vary from 90.15% to 100%, depending on the characteristics of ECG signal morphology as well as the noise. Analysis of records 105, 202, 213, and 221, for which classification performance was significantly below the average, revealed specific ECG signal characteristics, which limits its performance. The first limitation of the proposed algorithm are the multiform and late-cycle PVC beats, which frequently result in fusion PVCs. This was revealed by record 213 in large number of false negatives. The misclassification of such beats as normal is the result of morphology of the fusion PVCs, which varies from almost normal to almost identical to that of the PVCs. Multiform PVCs are also the

Record	Total (beats)	Abnormal (beats)	Normal (beats)	TP (beats)	FP (beats)	FN (beats)	TN (beats)	Acc (%)	Se (%)	Sp (%)
100	2273	34	2239	34	2	0	2237	99.91	100.00	99.91
103	2084	2	2082	2	34	0	2048	98.37	100.00	98.37
105	2555	45	2510	43	104	2	2406	95.85	95.56	95.86
111	2123	1	2122	1	23	0	2099	98.92	100.00	98.92
113	1794	6	1788	6	55	0	1733	96.93	100.00	96.92
117	1535	1	1534	1	19	0	1515	98.76	100.00	98.76
121	1862	2	1860	2	10	0	1850	99.46	100.00	99.46
123	1518	3	1515	3	7	0	1508	99.54	100.00	99.54
200	2597	855	1742	828	19	27	1723	98.23	96.84	98.91
202	2135	74	2061	62	108	12	1953	94.38	83.78	94.76
210	2648	225	2423	189	14	36	2409	98.11	84.00	99.42
212	2748	0	2748	0	3	0	2745	99.89	100.00	99.89
213	3250	609	2641	231	8	378	2633	88.12	37.93	99.70
214	2259	258	2001	253	1	5	2000	99.73	98.06	99.95
219	2154	72	2082	68	27	4	2055	98.56	94.44	98.70
221	2427	396	2031	152	0	244	2031	89.95	38.38	100.00
222	2480	207	2273	143	224	64	2049	88.39	69.08	90.15
228	2048	364	1684	362	18	2	1666	99.02	99.45	98.93
231	1571	3	1568	3	11	0	1557	99.24	100.00	99.30
232	1780	1382	398	1343	4	39	394	97.58	97.18	98.99
233	3078	849	2229	844	2	5	2227	99.77	99.41	99.91
234	2753	53	2700	46	2	7	2698	99.67	86.79	99.93
Total	49672	5441	44231	4616	695	825	43536	96.94	84.84	98.43

Table 7.12 Results of performance evaluation for the proposed SVM classification algorithm using MIT/BIH Database.

cause of poor performance on record 221. In this case however one subject specific PVC form was more common than in the other records, what did not allow the classifier to learn it. The other two records 105 and 202, for which larger than at average number of false positives was observed, contain an exceptionally large amount of high-grade noise, artefacts as well as exhibits various irregular rhythms. Thus, those normal, but noisy beats with irregular morphologies, were misclassified as abnormal, resulting in worse specificity, when compared with other, rather moderate noise contaminated, records.

However, as can be observed from Table 7.12, the performance of the proposed algorithm for all the remaining records, was near or above the average classification performance. The only exception is record 222 for which the likely reason for the low sensitivity is that there were only 16 nodal escape beat (j) classified as normal in the training data what is notably less than the 213 beats available in the record 222. Thus relatively small number of those beats in the training data set, did not allow the classifier to learn their pattern what led to misclassification of nodal escape beat (j) pattern as abnormal. Nevertheless, results presented here compare favourably with published results of other heart beat classification algorithms, as evidenced when comparing the proposed algorithm with other methods in Section 8.3.2.

7.7   Multi parameter signal analyses using unsupervised learning algorithms

Due to the complexity of the human body, the analysis of intrinsically correlated physiological signals is required in order to get more complete information about patient's health status. The multi-parameter medical monitoring (Anliker, Ward, Lukowicz *et al.* 2004) as well as multi-sensor data fusion (Kenneth, Rajendra Acharya, Kannathal *et al.* 2005) are the most promising techniques that could improve the robustness of a system and enable more unified clinical reasoning. There is a number of related works that use multi-parameter monitoring (Tarassenko, Hann, Patterson *et al.* 2005, Mundt, Montgomery, Udoh *et al.* 2005) along with data aggregation and fusion (Thoraval, Carrault, Schleich *et al.* 1997, Kannathal, Acharya, Ng *et al.* 2006) to reduce false alarms and provide higher accuracy. However, as recent research shows (Alemzadeh, Zhanpeng, Kalbarczyk *et al.* 2011) existing rule-based monitoring schemes, used for instance in ICU monitors, show significant limitations in terms of monitoring and classifying fused vital signs. Rule-based model's limitations are even more apparent, when these models are applied to context sensitive ubiquitous health monitoring applications (such as the one proposed in this work), where patient movements or environmental noise, amongst other artefacts leading to false alarms, are very common. Therefore, an increased interest is observed in patient-specific and adaptive monitoring as it has proved to be more effective in identifying the potential health risks and specific clinical symptoms of an individual, compared with the conventional population-based diagnostic flows (Y. H. Hu, S. Palreddy and W. J. Tompkins 1997, Zhang 2007).

Proposed in this chapter is an embedded, user adaptable and multi-parametric well-being analysis model based on Kohonen's Self Organising Map (SOM), which aims to improve medical monitoring of individual's health status during normal daily activities. The model combines traditional rule-based assessment of vital signs with the SOM's unsupervised ability to adapt over use to the patient. As a result it creates a two-dimensional physiological "health map" of a patient, where any deterioration from patient-specific optimal health status as well as any potential health risks can be detected and tracked through observation of the Best Matching Unit (BMU) over time. During the initial training phase the population-based "health map" is compiled by the aggregation of different vital signs from the representative examples of multi-parameter health analysis contained in MIMIC database (Moody and Mark 1996). Proposed techniques allow over time to tailor such population-based "health map" towards a patient-specific scheme, where the individual's personal physiological characteristics are considered for detecting deterioration and potential abnormalities. The SOM adaptation is controlled by parameter specific as well as aggregate score according to NEWS. The proposed model was tested on the remaining records of the MIMIC database. Obtained experimental results demonstrate the effectiveness of the proposed approach in masking false alarms and imperfections of the rule-based detection schemes.

7.7.1   Self-Organizing Map (SOM)

The Self-Organizing Map (SOM) belongs to the class of unsupervised and competitive learning algorithms. The SOM, is a sheet-like neural network, which uses a set of neurons (also referred to as nodes or units) arranged in a two-dimensional rectangular or hexagonal grid of size $M \times N$, to form a discrete topological mapping of an input space $X \in R^K$, as illustrated in Figure 7.28 and 7.29. Each node in the map has a specific position (i, j) coordinates and is directly associated with a weight vector of the same dimension as the input vector. The items in the input data set are assumed to be in a vector format. If K is the dimension of the input space, then every node on the map grid holds an K-dimensional vector of weights (91).

$$\mathbf{w}_{ij} = [\, \mathbf{w}_1 , \mathbf{w}_2 , \mathbf{w}_3 , .., \mathbf{w}_k \,] \tag{91}$$
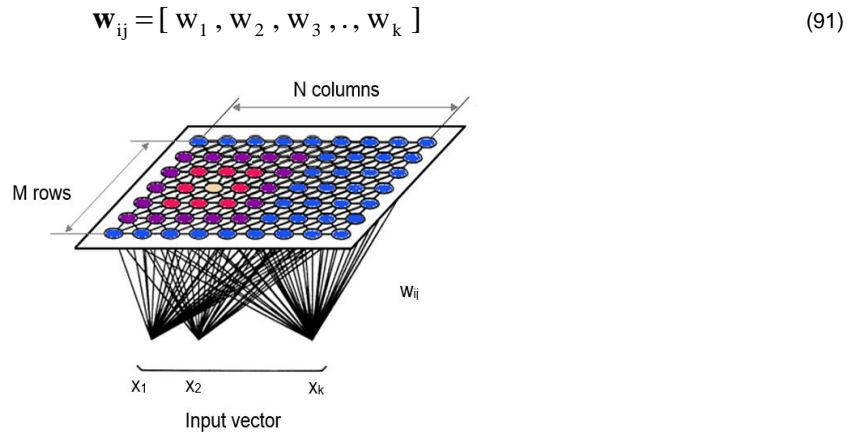
Figure 7.28 Self-Organising Map (SOM) topology.

The concept behind the Self-Organizing Map training rely on the iterative adjustment of these weight vectors until the map represents a picture of the input data set. Since the number of map nodes is significantly smaller than the number of items in the dataset, the objective is to achieve a configuration in which the distribution of the data is reflected and the most important metric relationships are preserved. Therefore the ultimate goal is to obtain a correlation between the similarity of items in the dataset and the distance of their most alike representatives on the map. In other words, items that are similar in the input space should map to nearby nodes on the grid.

Rectangular   Hexagonal
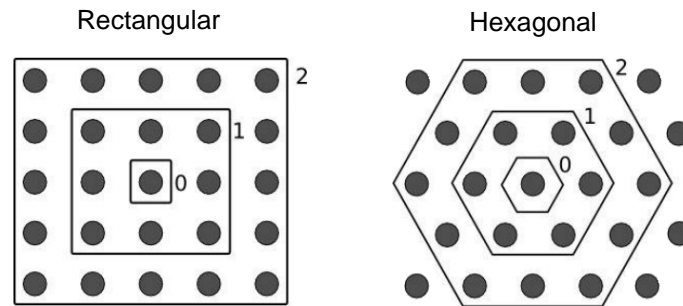
Figure 7.29 Types of grids with neighbourhood at radius r=1,2 around Best Matching Unit.

7.7.1.1   Map initialization

At the start of the learning process, all the weights must be initialized. It is an important first step in map design process as the performance of the SOM algorithm depends on the initial weights of the map. Initialization methods can broadly be classified as: 1) random or pattern based, 2) based on the random samples from the input training data set and 3) data analysis based initialization methods, called Principal Component Initialization (PCI) (Akinduko and Mirkes 2012). In the first approach weight are either initialized to small random numbers or numbers that follow a certain pattern, like corner or centre equidistant initialization illustrated in Figure 7.32, which are completely independent of the training data set. In this approach the initial map knows nothing about the input space. even though is simple, it requires additional training cycles so that the map could at least roughly represent the training data. The second approach use randomly selected samples from the training data set. In this case the advantage is that the initial weight vectors already lie in the same space as the training data, therefore when the training commences, the map is already in a state in which it represents at least a subset of the input data set, what reduce number of iterations. However, as the input samples used for the initialization are random, relatively small subset of the training data is used. Therefore two scenarios are likely to occur: 1) a large number of outliers, which have little in common with the majority of data items, might get chosen or 2) only part of the spectrum is represented. In such a case the initial map is not likely to be truly representative of the given dataset. Similarly, completely reversed situation is also likely to be true, where the initial samples may not reflect the existence of outliers at all.

The second approach ties to reflect the distribution of the data more faithfully by characterizing the dataset with those dimensions, which show the strongest variation. For this purposes the Principal Component Analysis (PCA) is used to reduce the dimensionality of the dataset, while retaining most of its original variability. It is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components, which number is less than or equal to the number of original variables. PCA is defined in such a way that the first principal component has the largest possible variance which accounts for as much of the variability in the data as possible, while each succeeding component in turn has the highest variance possible under the constraint it is uncorrelated with the preceding components.

PCA is based on concepts of covariance and eigenvectors. The covariance is a similar measure to variance but applied to two dimensions where it measures how much these two dimensions vary from the mean with respect to each other. The covariance of two features vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is defined as the average of the products of the deviations of features values from their means and its formula can be written as (Smith 2002):

$$\sigma_{ij} = \mathrm{cov}(\mathbf{x}_i, \mathbf{x}_j) = \frac{\sum_{k=1}^{K}(x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j)}{K-1} \tag{92}$$

The exact covariance value is not as important as the sign is. If the value is positive then that indicates that both dimensions increase together. If the value is negative, then as one dimension increases, the other decreases. For the latter case, if the covariance is zero, it indicates that the two dimensions are independent of each other. If we calculate the covariance for all n dimensions in our input data set as a result we get a square covariance matrix of size $n \times n$ as:

$$C = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{in} \\ \sigma_{21} & \sigma_{22} & & \sigma_{2n} \\ \vdots & & \ddots & \\ \sigma_{n1} & \sigma_{n2} & & \sigma_{nn} \end{bmatrix} \tag{93}$$

Covariance matrix is, as the covariance values on the diagonal are the variances for each dimension. Since $\sigma_{ij} = \sigma_{ji}$ it is also symmetrical about the main diagonal. This ensures that the second element of PCA, the eigenvector and eigenvalues, exists.

An eigenvector of a square matrix A is a non-zero vector x, with $x \in R^n$ such that when the matrix multiplies the vector it yields a scalar λ as follows:

$$\mathbf{Ax} = \lambda \mathbf{x} \tag{94}$$

The λ is called the eigenvalue of matrix $\mathbf{A}$ corresponding to $\mathbf{x}$. The $\mathbf{A}$ matrix can be thought of as a transformation matrix which multiplied on the left of a vector, gives another vector that is transformed from its original position. An eigenvector of $\mathbf{A}$ is a vector that gives a direction in which that transformation is simply a scaling transformation. The amount of scaling

is the associated eigenvalue. The larger the eigenvalue, the grater the variation along its associated eigenvector (Smith 2002). The eigenvectors of the covariance matrix, illustrated in Figure 7.30 on top of the scatter plot, indicates the directions in which data points are stretched.
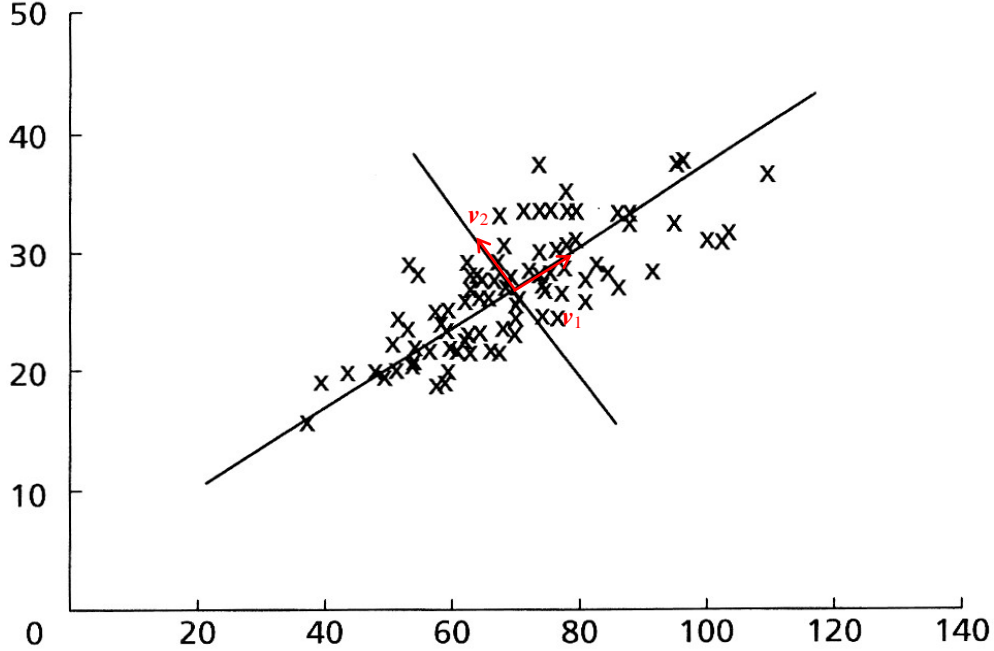
Figure 7.30 Sample data set characterised by the eigenvectors (Swan and Sandilands 1995).

The SOM initialisation procedure (Schatzmann and Ghanem 2003), used in this study, applies the above introduced concepts of Principle Component Analysis to the whole training dataset D, encoded as the $N \times K$ matrix, where K is the dimension of each input data item and N is the size of the training dataset. In the first step the mean vector $\bar{\mathbf{x}} = [\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_k]$ of the dataset is calculated, where $\bar{x}_i$ is the mean of the ith column (i^{th} dimension). The mean vector is further used in the second step to find the square, diagonal covariance matrix of size $K \times K$. This, in turn, allows obtaining the set of K eigenvalues, ordered according to their magnitudes from which the first two dominant eigenvectors are used to initialize the map.

The initialization starts by placing the mean $\bar{\mathbf{x}}$ weight vector at the centre of the map, whilst the two dominant eigenvectors $\mathbf{v}_1$ and $\mathbf{v}_2$ associated with λ_1 and λ_2 are used to scale the main axes as illustrated in Figure 7.31. Map nodes to the right and left of the mean are derived by respectively adding and subtracting multiples of the eigenvector $\mathbf{v}_1$. Similarly map nodes below and above the mean are found by respectively adding and subtracting multiples of $\mathbf{v}_2$. Map nodes that do not lie on the axes are determined by adding a linear combination of $\mathbf{v}_1$ and $\mathbf{v}_2$ (Schatzmann and Ghanem 2003).

$\mathbf{v}_1$

$\mathbf{v}_2$

Figure 7.31 Map Initialization using the Principal Component Analysis (Schatzmann and Ghanem 2003).

As highlighted in Figure 7.31 the borders of the map, denoted as p and q are the largest coefficients of the $\mathbf{v}_1$ and $\mathbf{v}_2$. In order to reflect the actual distribution of the data items, not their full spectrum, it is then reasonable to think of p and q as the borders of some interval containing a significant part of the dataset. The proposed limit on their values is expressed in terms of sample standard deviation of the data as $2s$ from the mean value, what covers a fair percentile of the data items without attributing too much importance to the outliers (Schatzmann and Ghanem 2003). Assuming N and M to be the width and height of the map, a general formula for weight vector $\mathbf{w}_{ij}$ of node (i, j) of the initial map can be formulated as:

$$\mathbf{w}_{ij} = \bar{x} + p_i \mathbf{v}_1 + q_j \mathbf{v}_2 \tag{95}$$

where $p_i = 2s \cdot \left(\dfrac{2i}{N} - 1 \right)$ and $q_j = 2s \cdot \left(\dfrac{2j}{M} - 1 \right)$

Figure 7.32 compares three Self-Organizing Maps, initialised using three different initialization algorithms – a) centre equidistant; b) randomly selected samples from the input training data (5% of training dataset); and c) initialization based on PCA of the whole training dataset.
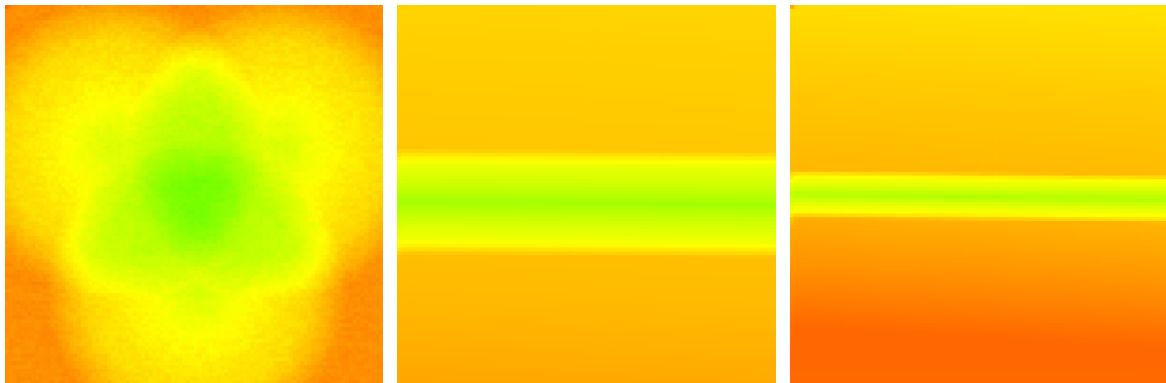
Figure 7.32 a) centre equidistant, b) randomly selected samples (5% of total DS1 samples), and c) PCA initialization.

7.7.1.2   The SOM learning algorithm

The SOM learning algorithm is an iterative process. At each discrete time index t a training data sample $\mathbf{x}(t)$ from the input space is selected and presented to the network. The whole learning process is competitive, meaning that the algorithm determines a winning node c on the map, whose weight vector $\mathbf{w}_c$ is most similar to the input sample $\mathbf{x}$ and modify this node weight vector along with its close neighbourhood. The whole process is shown in the listing below. For the sake of simplicity, let's consider the map as 1-dimensional vector of weights vectors $\mathbf{w}_i$, where each vector is of the same dimension K as the input dataset and let N be the total number of those neurons. The conversion from 2D matrix to 1D vector is illustrated in Figure 7.33.
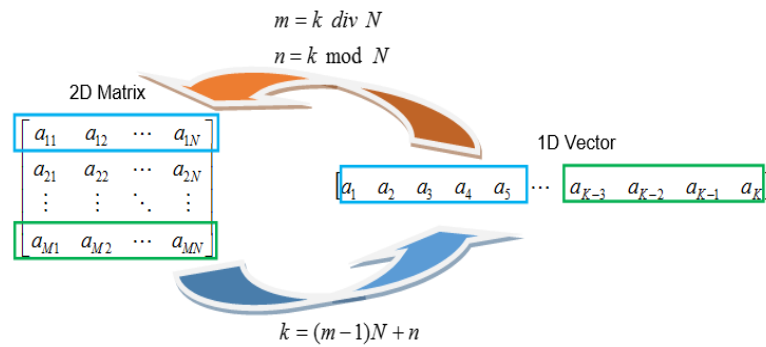
Figure 7.33 Conversion of 2D matrix to 1D vector.

1) Initialization: An initial weight is assigned to all the nodes

repeat

  2) Competition: At each discrete time index t, present an input **x**(t) and select the winner node c according to Euclidian distance criterion:

$$c(x) = \arg\min_{i} \left\| \mathbf{x}(t) - \mathbf{w}_i(t) \right\| \quad, i = 1, 2, ..., N \tag{96}$$

  3) Cooperation: the winning neuron excites its neighbouring neurons according to Gaussian neighbourhood function as:

$$\eta_{ci}(t) = \alpha(t) \exp\left(-\frac{\left\| r_c - r_i \right\|^2}{2\sigma^2(t)} \right) \tag{97}$$

  4) Learning Process: updates the weights of the winner and its neighbours, with the rule given below:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \eta_{ci}(t)\left[\mathbf{x}(t) - \mathbf{w}_i(t) \right] \tag{98}$$

until the map converges

At the start of the learning, all the weights gets their initial value according to the selected initialization algorithm, as discussed in previous section. The iterative part of the algorithm starts

with the competition phase, in which all nodes compete for the ownership of the input pattern. Using the Euclidean distance as criterion (96), the neuron with the minimum-distance wins. In the next phase, called cooperation, the winning neuron excites its neighbour neurons, which are topologically close. A neighbourhood function often used is the Gaussian neighbourhood function given by equation (97), where $\alpha(t)$ is a time dependent parameter called the learning rate. It may be regarded as the height of the neighbourhood function, which controls the speed of adaptation. The second variable parameter $\sigma(t)$ represents the effective range of the neighbourhood function. It may be considered as width or radius of the neighbourhood function, which specifies the region of influence that the input sample has on the map. Such Gaussian curve, decreasing from the neighbourhood centre node c to the outer limits of the neighbourhood is illustrated in Figure 7.34.
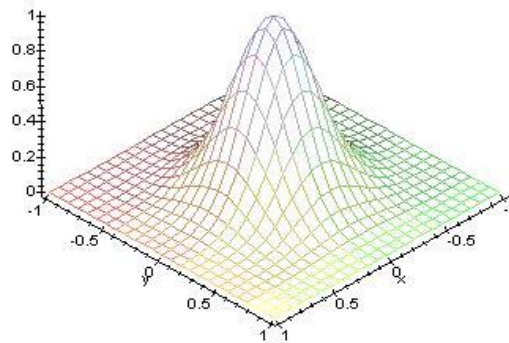
Figure 7.34 2-D Gaussian distribution with mean (0,0) and σ=1.

Finally, in the learning phase the winning neuron and its neighbours are adjusted with the rule given in equation (98). Hence, the weight of the winning neuron and its neighbours are adjusted towards the input patterns, however the neighbours have their weights adjusted with a value less than the winning neuron. This helps to preserve the topology of the map. The whole process repeats until the map converges.

7.7.1.3   Patient adaptation algorithm

For traditional SOM algorithms both the height and the width of the neighbourhood function decrease monotonically with time. At the beginning of the learning process, the best matching unit will be modified very strongly and the neighbourhood is fairly large. However, towards the end of the learning process, only very slight modifications will take place and the neighbourhood will include little more than the BMU itself. This corresponds to "rough ordering" at the beginning of the training phase and "fine tuning" near the end. This is a desired operation for the initial training of the population-based Health Map, however become a problem in terms of user adaptation over use. This is because such converged, population trained SOM algorithm, cannot learn with adequate speed new incoming patient-specific samples that may be different in statistical characteristic from the previously learnt population-based samples. In

other words, the learning process is incapable of responding appropriately to a varied environment that embodies incoming samples (Heskes and Kappen 1991).

To resolve the aforementioned problem, the suggestion is to choose time–independent learning parameters, which change their values according to the conditions of the incoming samples and not to the elapse of time. These should increase the capability of SOM in dealing with varied environments, but should not decrease the speed of convergence of the SOM algorithm, what is the case with constant learning rates. These should ensure that the algorithm stay stable in such varied environments. Therefore traditionally trained on population samples the Self-Organising Map, applied to ubiquities patient monitoring adapts over time to the patient according to the Time Adaptive Self-Organising Map (TASOM) algorithm proposed by Shah-Hosseini and Safabakhsh (2000). TASOM automatically adjust the learning parameters and incorporates possible changes of the input distribution in updating the synaptic weight. For this purpose, the learning rate of each neuron is considered to follow the values of a function of distance between the input vector and its synaptic weight vector. Such parameter will be changed independently for each node and the number of these parameters will be equal to the number of nodes on the map. A similar updating rule is proposed to automatically adjust the neighbourhood size of each node. It is considered to follow the distance between the node's weight vector and the weight vector of its neighbouring nodes. For patient adaptation, the learning-rate parameters $\eta_i(0)$ of all nodes should be initialized with values close to unity, while neighbourhood function $\sigma_i(0)$ should be initialized to include all nodes within the cluster. Other constant parameters $\alpha, \beta, \alpha_s, \beta_s$ as well as $s_k(0), E_k(0), E2_k(0)$ components of the scaling vector can have any values between zero and one. The detailed algorithm is presented in the listing below.

```
for each new input x(t) where NEWS[x(t)] <= 3 do
    1) Competition: select the winner node c at time t, using the
       minimum-distance Euclidean norm as the matching measure as:
```

$$c(\mathbf{x}) = \arg\min_i \left\| \mathbf{x}(t) - \mathbf{w}_i(t) \right\|_s, \quad i = 1,2,...,N \tag{99}$$

```
    where
```

$$\left\| \mathbf{x}(t) - \mathbf{w}_i(t) \right\|_s = \sqrt{\sum_k \left(\left(x_k(t) - w_{j,k}(t) \right) / s_k(t) \right)^2} \tag{100}$$

```
    2) Update the neighbourhood size: Adjust the neighbourhood width
       σ_c(t+1) of the winning node c by the following equation:
```

$$\sigma_c(t+1) = \sigma_c(t) + \beta \left[g \left(\frac{1}{\#(NH_c) \cdot s_g} \sum_{i \in NH_c} \left\| \mathbf{w}_c(t) - \mathbf{w}_i(t) \right\|_s \right) - \sigma_c(t) \right] \tag{101}$$

```
    where the function #(.) gives the cardinality of a set, while
    NH_i is the set of neighbouring nodes of any neuron i in the
```

lattice. For the one-dimensional lattice N, used in this
project, NH$_i$={i-1,i+1}, where NH$_N$ ={N-1} and NH$_1$={2}.

3) Update the set of neighbour nodes: Select the neighbour nodes
   around the winning node c for further processing as:

$$\Lambda_c(t+1) = \left\{ i \in N \middle| d(c,i) \leq \sigma_c(t+1) \right\} \tag{102}$$

where $d(c,i)$ is the distance between two nodes on the grid.

4) Update the learning rate parameters: Adjust the learning rate
   parameters η$_i$(t+1) in the neighbourhood Λ_c of the winning node:

$$\eta_i(t+1) = \eta_i(t) + \alpha \left[f\left(\frac{\|\mathbf{x}(t) - \mathbf{w}_i(t)\|_s}{s_f} \right) - \eta_i(t) \right] \quad for\ i \in \Lambda_c(t+1) \tag{103}$$

5) Learning Process: updates the weights of the winner and its
   neighbours, with the rule given below:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \eta_i(t+1)\left[\mathbf{x}(t) - \mathbf{w}_i(t)\right] \quad for\ i \in \Lambda_c(t+1) \tag{104}$$

6) Update the input scaling vector: Adjust the scaling vector
   s$_k$(t+1) for all K features of the input vector **x**(t)

$$s_k(t+1) = \begin{cases} \sqrt{E2_k(t+1) - E_k(t+1)^2} & if\ \left(E2_k(t+1) - E_k(t+1)^2\right) > 0 \\ 0 & otherwise \end{cases} \tag{105}$$

where:

$$E2_k(t+1) = E2_k(t) + \alpha_s\left(x_k^2(t) - E2_k(t)\right)$$

$$E_k(t+1) = E_k(t) + \beta_s\left(x_k(t) - E_k(t)\right)$$

End

Following (Shah-Hosseini and Safabakhsh 2000) monotonically increasing scalar functions $f(\cdot)$, used for normalization of the distance between the weight and input vectors and $g(\cdot)$ used for normalization of the weight distances, are $f(z) = z/(z+1)$ and $g(z) = (N-1)(z/(z+1))$. The constant parameters s_f and s_g, which are the slopes of the functions $f(\cdot)$ and $g(\cdot)$, respectively, are application specific and are determined experimentally using a given training dataset. The parameter s_f controls the trade-off between generalization and memorization. When s_f is low, the TASOM weights are specialized to approximate the most recent data and forget the previous information with higher speed. On the contrary, when s_f is high, forgetting the previous information is done more slowly, and thus generalization ability of the TASOM is increased. In turn, the s_g parameter controls the compactness and topological ordering of TASOM weights When s_g is low, the weights remain

close to each other what preserves the topological ordering and the weights cluster around the centre of the input distribution. As s_g increase, the weights become looser, and better approximated the input distribution specialized to approximate the most recent data and forget the previous information with higher speed. Empirical analyses of the TASOM operation on representative validation records 212 and 216 of the DS1 data set, indicated a low value of $s_g = 1$ in order to focus on a central value or a typical value (central tendency) in a distribution of each vital sign, as well as high value of $s_f = 10$ in order to focus in adaptation on the longer rather short term changes in health status.

7.7.1.4   Map visualization

The visual representation of the Kohonen map provides a synthetic and meaningful representation for the temporal evolution of individual physiological signals as well as the overall integrative health status. There are two basic types of visualizations: component planes and single planes. The component planes show each dimension of the data on a separate map. The single plane visualisations show all dimensions in one combined map. Both visualization techniques use colours to show the cluster structure of the map. For this purpose the visualization algorithm allocates similar colours to nodes with similar weight vectors on the map. Once every node is represented by a block of coloured pixels it is easy to detect groupings and outliers.

Effective and meaningful visualisation method requires achievement of some correspondence between a node's colour and its weight vector. Let's consider, for example, the most popular RGB colour scheme, defined by three colour components red, green, and blue as illustrated in Figure 7.35a. It is commonly used for the sensing, representation, and display of images in electronic systems. By using this colour scheme each weight vector dimension could simply be mapped onto one colour component. The obvious limitation of such mapping is that only up to 3-dimensional weight vectors could be represented in this way. In practice, however, weight vectors are usually of much higher dimension (often not even known beforehand) then the dimension of any colour space. Another issue is the meaning of colours used, which should be as intuitive as possible for the human eye and the application. One of the most popular colour coding systems across various applications is so the called traffic light or RAG (Red Amber Green) report system. It is a very elegant idea, which gives a clear indication of whether things work or not in a simple, highly visual way. Therefore application of such well understood model to colour code the self-organizing maps for health monitoring, by associating red with abnormality (high-risk), amber/yellow with anomaly (medium-risk) and green with optimal health status (low-risk), seems as the right and most intuitive solution.
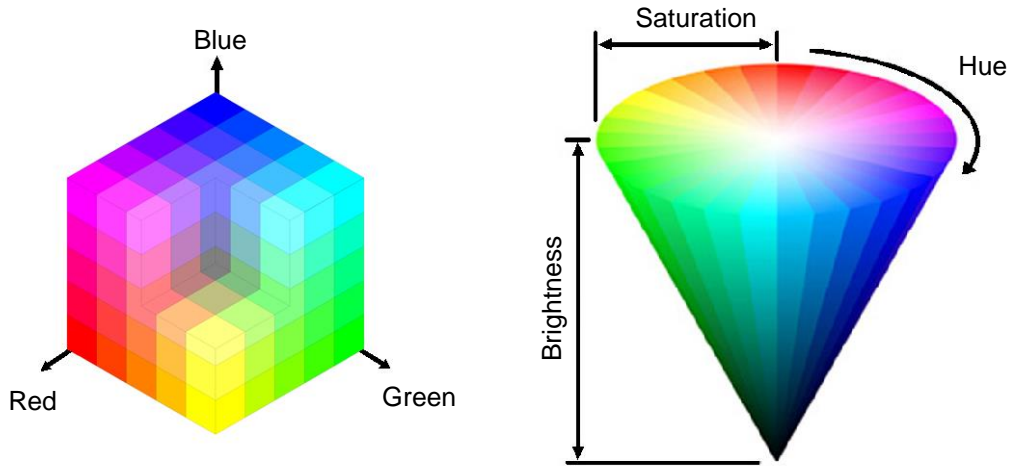
Figure 7.35 a) The RGB colour scheme; b) The HSB colour scheme

The map visualisation method, proposed in this work, makes use of the HSB colour scheme. The HSB references colours in terms of their Hue, Saturation and Brightness. Each of these three features takes on a floating point value between 0 and 1. The easiest way to visualize this scheme is to think of the H, S, and B values representing points within an upside-down cone as illustrated in Figure 7.35b. From the point of view of this method the most important element of the HSB model is the hue parameter, which describes the actual colour. It is measured in angular degrees counter-clockwise around the cone starting and ending at red = 0^0 (hue = 0), or 360^0 (hue = 1), Here, the hue corresponds to the angle of a slice on the colour wheel, with yellow at 60^0, green at 120^0, etc. The last two parameters are, Saturation which is the purity of the colour, measured in percent from the centre of the cone (0) to the surface (1), and Brightness of the colour, measured in percent from top of the cone black (0) to the bottom white (1). For the proposed algorithm, both saturation and brightness are set to 1, in order to use only pure colours from the outer edge of the top of the cone.

Using the HSB colour scheme presented above, the algorithm proposes the colour code as a set of n-dimensional vectors which give n different colours by cutting the colour wheel at n equally spaced angles. In order to colour the map according to RAG report system the initial colour wheel is reduced here to the angle of 120^0 (hue = 0.33) and then cut at 3 equally spaced pieces that corresponds to red, amber and green colours, respectively. For this algorithm to work correctly, the map needs to be normalised according to NEWS model. In that way, each of the n-dimensions for which signals are within their normal ranges, have values ranging from -1 to 1. For signals that are outside of their normal ranges these values are therefore below -1 or above 1. In order to normalise the value of each dimension to 0-1 range and therefore be able to use RAG colour scale as 0 – green, 0.5 – amber, 1 – red, the signal specific scaling functions, defined in equation (106) are used. In that way, the individual characteristic of each vital signs' accepted ranges can be represented.

232

$$S_{v_i} = \begin{cases} \frac{1}{2}\log_2\left(|v_i|+1\right) & if \quad v_i \in [-1;1] \\ 0.5 + \log_a\left(|v_i|\right) & otherwise \end{cases} \qquad (106)$$

The a parameter in equation (105) is the dimension dependent scaling factor, which controls how fast low-risk (green) converts to medium-risk (amber) and then to high-risk (red). Such obtained colour scale allows in the next step to pick the right colour, from the HSB colour scheme. This is done individually for each dimension according to a simple linear formula (107) in order to produce component planes.

$$C_{v_i} = -0.33S_{v_i} + 0.33 \qquad (107)$$
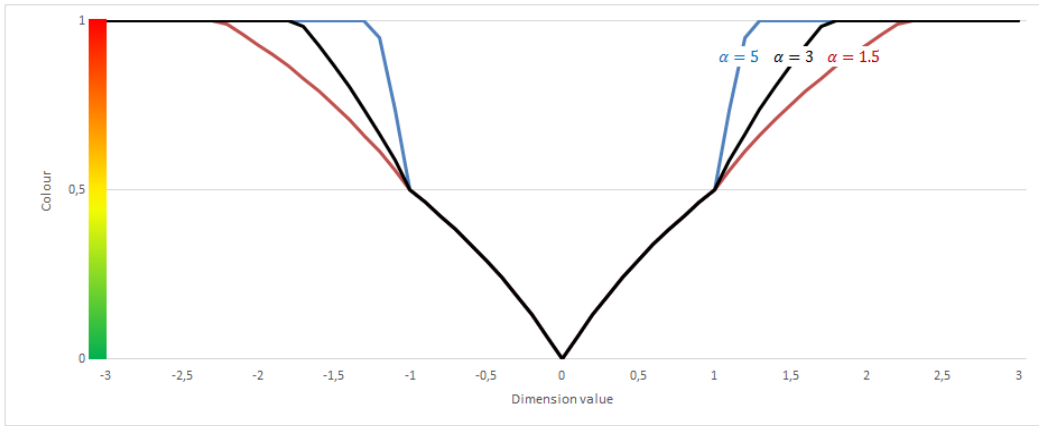
Figure 7.36 Scaling function that maps individual component's value to RAG colour scale.

Finally, when all the individual component planes are known, the single plane can be produced, by calculating single colour $C_{\mathbf{v}}$ for each node on the map, as the average of component colours for its n-dimension weight vector, as defined in equation (108).

$$C_{\mathbf{v}} = \frac{\sum\limits_{i=1}^{n} C_{v_i}}{n} \qquad (108)$$

7.7.2 The multi-parameter health status classification

Development and evaluation of automated multi-parameter health decision support systems requires large amount of well-characterized physiological data. In order to refine the proposed algorithm, their training and testing with realistic data, as well as performing these tests repeatedly and reproducibly, is essential. To achieve this aim, this project uses the Multi-parameter Intelligent Monitoring for Intensive Care (MIMIC) database (Moody and Mark 1996) to simulate patients during development and evaluation of intelligent monitors. The MIMIC database includes in total 121 records, obtained from over 90 medical, surgical, and cardiac patients of Boston's Beth Israel Hospital. The data in each case include signals and periodic measurements obtained from a bedside monitor as well as clinical data obtained from the patient's medical record. The recordings vary in length; almost all of them are at least 20 hours,

and many are 40 hours or more. In all, the database contains nearly 200 patient-days of real-time signals and accompanying data. What is also important, the MIMIC database includes comprehensive data from each subject's medical record, providing context for each recording. Although existing ICU monitors do not often make use of these data, it is clear that any process intended to provide competent support for medical decisions must be aware of all of what is known about the patient. This it is thus regarded as an essential component used during the development and testing of the proposed model to evenly distribute data between training and testing datasets.

Typical MIMIC continuous record contains two or three ECG signals, each sampled at 500 Hz, and four or five other signals, usually including arterial blood pressure (ABP), respiration (RESP), and pulse oximeter signal (SpO_2), each sampled at 125 Hz in addition to the monitor's periodic measurements, alarms, and monitor status messages. For the purposes of developing and testing the Health Map model the monitor's periodic measurements (referred to as "numerics" hereafter) are used in this study. These are slowly-changing variables recorded at intervals of 1.024 seconds that typically appear in numeric form on the ICU monitors' screens such as heart rate, blood pressure (mean, systolic, diastolic), respiration rate, oxygen saturation, etc.. Numerics make a perfect match with the parameters used in NEWS health assessment model adopted as the reference point in this study.

With the exception of the first 5 cases (records: 032, 033, 037, 039, and 041) all other records are accompanied by the patient status and monitoring condition alarms. Patient status alarms refer to events that (as determined by the ICU monitor) require medical intervention (for example, observation of heart rate or blood pressure outside of pre-set limits). Monitoring condition alarms refer to events that interfere with the function of the monitor (for example, transducer malfunction or signal saturation). Note that these alarms are as reported by the ICU monitors, including all false alarms. Once an alarm condition is detected, the monitors produce alarms at intervals of 1.024 seconds (0.976563 Hz) until reset by the ICU staff, or until the variable that triggered the alarm returns to its nominal range. There are also significant events in some cases that did not trigger alarms and are therefore not annotated.

The choice of MIMIC Database as the dataset for this research project is dictated by its particular usefulness in investigations of heart rate, blood pressure, and respiratory dynamics and their interactions, what is also the main focus of NEWS scoring system and the proposed model. It should be however noted that the MIMIC database was designed with the hemodynamically unstable patients in mind. Therefore it should be considered as the representative of the full range of pathophysiologies that result in sudden blood pressure, but it does not represent the entire ICU population, which is very diverse and its representation is difficult and time consuming. Therefore the algorithm and experimental results obtained with the MIMIC database should not be considered as the ultimate "product" but rather a proof of concept showing its capabilities. Future development of this algorithm requires recordings which

represent other significant groups of patients whose problems also require rapid assessment and appropriate intervention.

Since the data-gathering protocol was designed to have minimal impact on patient monitoring or care, the selection of measured variables varies among these records, according to the requirements of the ICU staff for appropriate care of the patients in each case. Therefore among all 121 numerics records only 54 records from 41 subjects were considered as complete records, i.e. they contain simultaneous recordings of arterial blood pressure (ABP), heart rate (HR), respiration (RESP) and pulse oximeter signal (SpO$_2$), which are compulsory parameters for the proposed model. Patients and records, presented in table 7.13a, that were included in this group are aged between 27 and 92 years old and represent such clinical classes as: CHF/pulmonary edema (11 patients / 14 records), MI/cardiogenic shock (6 patients / 6 records), Respiratory failure (8 patients / 12 records), Brain injury (2 patients / 3 records), Angina (2 patients / 2 records), Sepsis (2 patients / 3 records), Bleed (2 patients / 3 records), MI/arrest (1 patient / 1 record), Renal failure (1 patient / 1 record), Post-op valve (2 patients / 3 records), Post-op CABG (3 patients / 3 records), and No clinical class denoted as (-) (1 patient / 1 record). From this group four classes were removed: 1) the MI/arrest, Renal failure and No Clinical patients due to the insufficient number of records that could well represent data distribution in both training and testing datasets; and 2) two post-surgical representatives, Post-op valve and Post-op CABG patients, that could potentially introduce pathological conditions, which results from a surgical intervention and not a classical clinical condition. Moreover, records 472, which was inconsistent in terms of number of numeric records and time of recording according to medical records for this patient as well as record 220 which was relatively short comparing to other records in this database were remove. Finally, in order to balance well the training and testing data set two records 414, 252 were removed. Removal of these two records did not affect the distribution of clinical classes in the dataset as records those were obtained from subjects already represented by other record in the dataset. As a result, the table 7.13b presents the final dataset, containing 41 records obtained from 31 patients, which was divided into two datasets. The training dataset (DS1) contains numerics in 22 recordings from 21 patients, while testing dataset (DS2) contains numerics in 19 recordings obtained from 18 patients with the same approximate proportion of clinical classes and total recording times. Note that some subjects have been recorded more than once. In such cases their recordings were included in both DS1 and DS2 datasets. Similar to ECG arrhythmia classification the first dataset (DS1) is used for both training and validation of the model. The validation in terms of SOM algorithm focuses on detection of normal and abnormal clusters on the resulting map. The second dataset (DS2) is used for a final performance evaluation and comparison with ICU monitors' alarms.

a)

	Clinical Class												Total
No. Of. Sub.	11	6	8	2	2	2	2	1	2	1	3	1	41
No. Of. Rec.	14	6	12	3	4	3	3	1	3	1	3	1	54
	CHF/pulmonary edema	MI/cardiogenic shock	Respiratory failure	Brain injury	Angina	Sepsis	Bleed	MI/arrest	Post-op valve	Renal failure	Post-op CABG	-	
Records	212	232	55	220,221	240	269,271	408,409	427	442	471	455	474	
	213	235	211	449	267,268,467	291	472		464,465		457		
	214	237	216								476		
	230,231	248	226										
	245	264	241,242										
	259,260	293	243										
	413,414		252,253,254										
	415		401,403										
	417												
	418												
	466												

b)

	Clinical Class [no. of samples]																					Total
	CHF/pulmonary edema			MI/cardiogenic shock			Respiratory failure			Brain injury			Angina			Sepsis			Bleed			
No. Of. Sub.	10			6			8			2			2			2			1			31
No. Of. Rec.	13			6			11			2			4			3			2			41
Subjects	Records	DS1	DS2	Records	DS1	DS2	Records	DS1	DS2	Records	DS1[h]	DS2	Records	DS1	DS2	Records	DS1	DS2	Records	DS1	DS2	
	212	145 275		232		92 799	55		131 250	221	83 362		240		107 226	269	151 700		408	169 814		
	213		171 093	235	166 999		211		75 991	449		149 433	267	161 541		271		110 884	409		152 343	
	214	90 157		237	150 000		216	95 654					268	130 155		291	75 788					
	230		66 796	248		120 116	226	112 500					467*		144 930							
	231	150 000		264	162 346		241		70 163													
	245	150 585		293		176 367	242	149 414														
	259		165 224				243	164 063														
	260	46 289					253		149 414													
	413	75 124					254		149 415													
	415		147 655				401	88 387														
	417	42 773					403		149 999													
	418	95 162																				
	466		241 406																			
Total Time [h]		795 365	792 174		479 345	389 282		610 018	726 232		83 362	149 433		291 696	252 156		227 488	110 884		169 814	152 343	
Total	5 229 592																					
Total DS1	2 657 088	50,81%																				
Total DS2	2 572 504	49,19%																				

*) All samples after sample no. 144930 have been removed due to corrupted data

Table 7.13 a) Summary of all compatible records and, b) selected MIMIC database records split into two sets used for training/validation (DS1) and testing (DS2).

236

7.7.3 Health Map algorithm overview

As outlined in the first chapter, the goal of this project is to investigate the use of Self-Organizing Maps in adaptive health monitoring. Therefore part of the project objectives was to design, implement and evaluate a system that is capable of generating, visualising and analysing health maps as a mean to analyse health status of a human being. The proposed adaptive health monitoring based on self-organizing map algorithm, consists of three main stages: 1) a normalization stage, 2) a clustering stage, and 3) a visualization stage, which are graphically depicted in Figure 7.37.
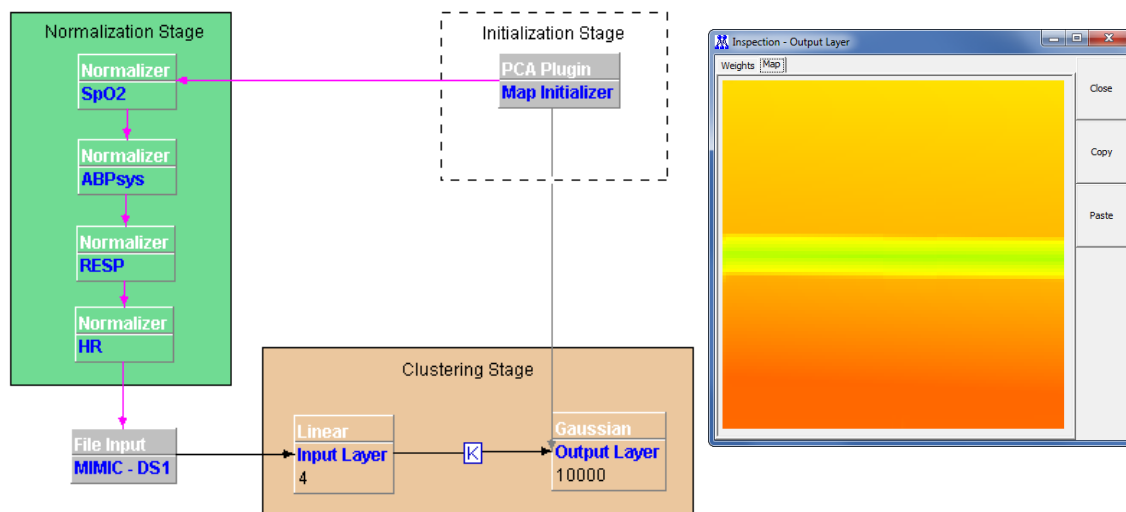
Figure 7.37 Block diagram of SOM classification algorithm in jMENN Editor.

In the normalization stage, each dimension of the input data is individually transformed to have a zero mean and standard deviation 1 within the [-1; 1] range, for samples within normal ranges defined in NEWS scales. In this way each dimension of the input pattern that has value between -1 and 1 can be considered as normal vital signs, while values above or below are considered as abnormal vital signs. This prevent one attribute from overpowering other attributes when clustering the population-based physiological signature of a patient.

Once the training data have been normalized, the clustering stage can begin. It is made of three elements: 1) the Linear layer, 2) the Kohonen synapse, and 3) the Gaussian layer. The Linear layer is the simplest kind of layer, which transfers the input pattern to the output applying a linear transformation, i.e. multiplying it by a constant term, called the beta term. Because beta term, in this case equal to 1, then the input pattern is transferred without modifications. The Linear layer is used here as a input layer, placed as the first layer of a network that permits to send an unmodified copy of the input pattern to the next component, which is the Kohonen synapse. The Kohonen synapse connects all the nodes of a linear layer with all the nodes of the Gaussian layer, permitting a pattern to be passed from one layer to another. It is the central element of the SOM, which holds the weight vectors for all nodes on the map. During the training phase, these weights are adjusted to map the N-dimensional input patterns to the two

237

dimensional map represented by Gaussian layer. Because it is both, the output synapse of a linear layer and the input synapse of the Gaussian layer, it represents a shared resource between those two layers. Hence to avoid a layer trying to read the pattern from its input synapse before the other layer has written it, the shared synapse in synchronized with the semaphore based mechanism. The last element of the SOM network is the Gaussian layer, which is the output layer of the SOM algorithm that implements the Gaussian Neighbourhood SOM strategy. It receives the Euclidean distances between the previous synapse (the Kohonen Synapse) weights and its input, and calculates the distance fall off between the winning node and all other nodes on the map. The distance fall off is calculated according to a Gaussian distribution from the winning node. This is then passed back to the Kohonen Synapse allowing it to adjust the weights feeding the winner neuron as well as weights of its neighbour, with a strength inversely proportional to the distance from the winner neuron. The speed as well as the number of neurons participating in the adaptation process is controlled by both the learning rate as well as the distance from the winner node, referred to as neighbourhood size or Gaussian size. However, before the first sample can be presented to the SOM algorithm, the initialization stage of the map have to be done. This is accomplished by the Principal Component Analysis (PCA) plug-in, which base on the whole training data set, produces the covariance matrix and determines the mean, eigenvector and the corresponding eigenvalue for each dimension of the input vector, as explained in details in Section 7.7.1.1.

At the beginning of the initial population-based training process, the algorithm maintains the neighbourhood size and the learning rate large in order to permit a large number of weights to participate to the adjustments with appropriate sensitivity to changes in the input pattern. This phase is named ordering phase, after which both learning rate and neighbourhood size decrease in order to eventually become very small, causing weights to freeze after they have chosen the input vectors to which to respond. At this point the map converges and the training process finishes. The speed of decrease is associated with the number of training epochs, it is how many times the total number of the input patterns is presented to the network. The epoch, the learning rate and the Gaussian size are the three main parameters that impact on the SOM algorithm ability to learn from the training patterns.

Once the training phase of the SOM neural network is completed, the output grid/map of neurons which is now stable to network iterations, can be visualized as one single plane or as n-dimensional component planes, according to SOM visualisation techniques presented in Section 7.7.1.4.The single plane map coloured according to the values of the four vital signs variables is useful in overall health assessment and to issue any health risk alarms. By analysing the single plane representation it can be clearly observed if the organization of the map is successful by checking weather all variables (colours) are smoothly varying on the map or not. In turn, the component planes representation allows to visualize the relative component distributions of the input data. Component plane representation can be thought as a sliced version of the Self-Organizing Map. Each component plane has the relative distribution of one

data vector component. Analysis of the component planes can give an indication if and how well did the SOM algorithm learn to recognise each vital sign's ranges and allows to spot redundant variables. If all colours are smoothly varying on the map for a given component then it can been considered as successfully clustered and not redundant. By comparing two or more component planes it can also be seen when two components correlate. If the outlook is similar, the components strongly correlate. Results of this analysis as well as the ability of the SOM algorithm to detect health risks are discussed in the following section.

7.7.4 Experimental results

The objective of this experiment was to show that a generic adaptive health monitoring system, based on the SOM algorithm presented in previous section, is able to discover the cluster relationships hidden in the vital signs data, with little or no a priori information about the problem domain. There is no intention to replace the doctor in diagnosing, but to support people in maintaining good health and recognise changes/health risks in physical parameters occurring during normal life. Therefore, the experiment was designed, to train the Self-Organizing Map on DS1 dataset of the MIMIC database, containing 2 657 088 records (approximately 730.08 hours of continuous recordings from ICU) and test its ability to discover health risks on DS2 dataset from the same database. For this purpose the ICU patients were considered as those whose vital signs showed significant variations (including pathological events) over a relatively short time spans. Using four numeric vital signs as variables: Heart Rate (HR), Respiratory Rate (RESP), Arterial Blood Pressure (ABP) and Pulse Oximetry (SpO2), an adaptive and integrative health status classification to low-, medium- and high-risk was performed using the green, amber and red colour codes, in order to assess the health status of an individual and issue alarms, when the input pattern gets classified as a high-risk area on the map.

The maps produced with the SOM algorithm are highly influenced by the choice of initial learning parameters. This includes:
4) the map width and height,
5) the number of iterations (epoch),
6) the size of the initial Gaussian neighbourhood and its ordering phase,
7) the initial value of the learning rate.

Because there are no strict guidelines for the selection of any of these parameters, a process of "trial and error" was adopted to determine a set of values that are suitable for the dataset at hand. In order to simplify the model selection process, a 10% training and validation subset of the DS1 data set was randomly selected and used to determine the most influential learning parameter which turned out to be the number of iterations (epoch). Therefore, the final model selection has been limited to the choice of appropriate number of epoch with all the other training parameters being set to the constant values. As such for all training trials using the full DS1 dataset, a 100 by 100 map was generated and its initial weights vectors were initialised

using Principal Component Analysis technique presented in section 7.7.1.1. During the initial model selection process it has been found that large maps and long training runs serve to smooth the map nodes significantly and reduce the average quantization error. The initial learning rate, which affects the amount of adaptation for the whole network, was set to 0.1 for all trials. There are at least two reasons why such a value was selected. Firstly, because of the large number of samples in the DS1 data set, and secondly, due to the PCA map initialization, which allowed to resemble the final topology of the map more closely, even before the actual train process begun. The initial neighbourhood size (Gaussian radius), for this problem was set to 10. Over the training process the neighbourhood size becomes altered through two phases: an ordering phase and a tuning phase. During the ordering phase the algorithm adjusts the neighbourhood size from the initial value down to 1, what allows neuron weights order themselves in the input space consistent with the associated neuron positions. At the beginning of the tuning phase, the neighbourhood size is less than 1. During this phase, the weights are expected to spread out relatively evenly over the input space while retaining their topological order found during the ordering phase.

With such settings in place the final model selection was performed on the growing sequences of 5, 10 and 20 epochs. The results of such training process are presented in Figures 7.38-40 below using three different views: the single plane RAG map, the clustered 3D map, and component planes for each dimension of the weights vector.
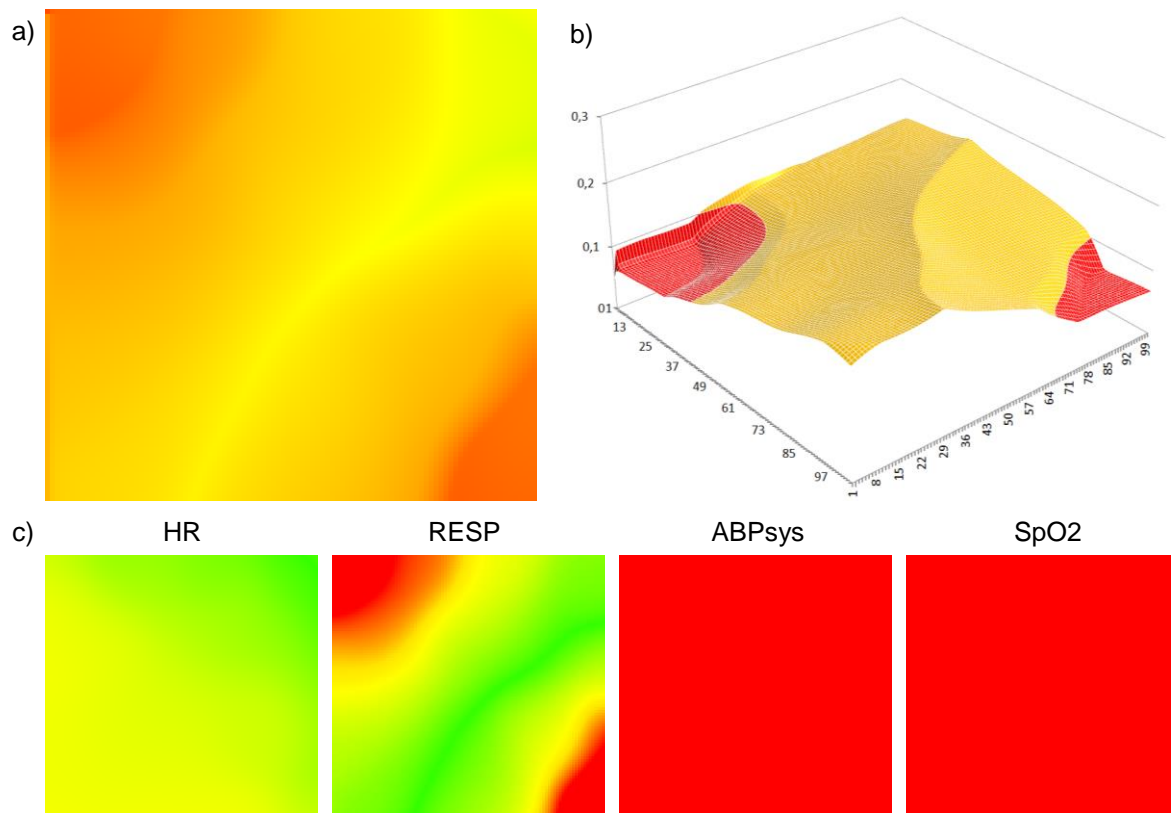
Figure 7.38 a) The single plane RAG map, b) the clustered 3D map, and c) the component planes view for each vital sign for the Self-Organizing Map formed after 5 epochs of the full DS1 dataset.

a)

b)

c)

| HR | RESP | ABPsys | SpO2 |
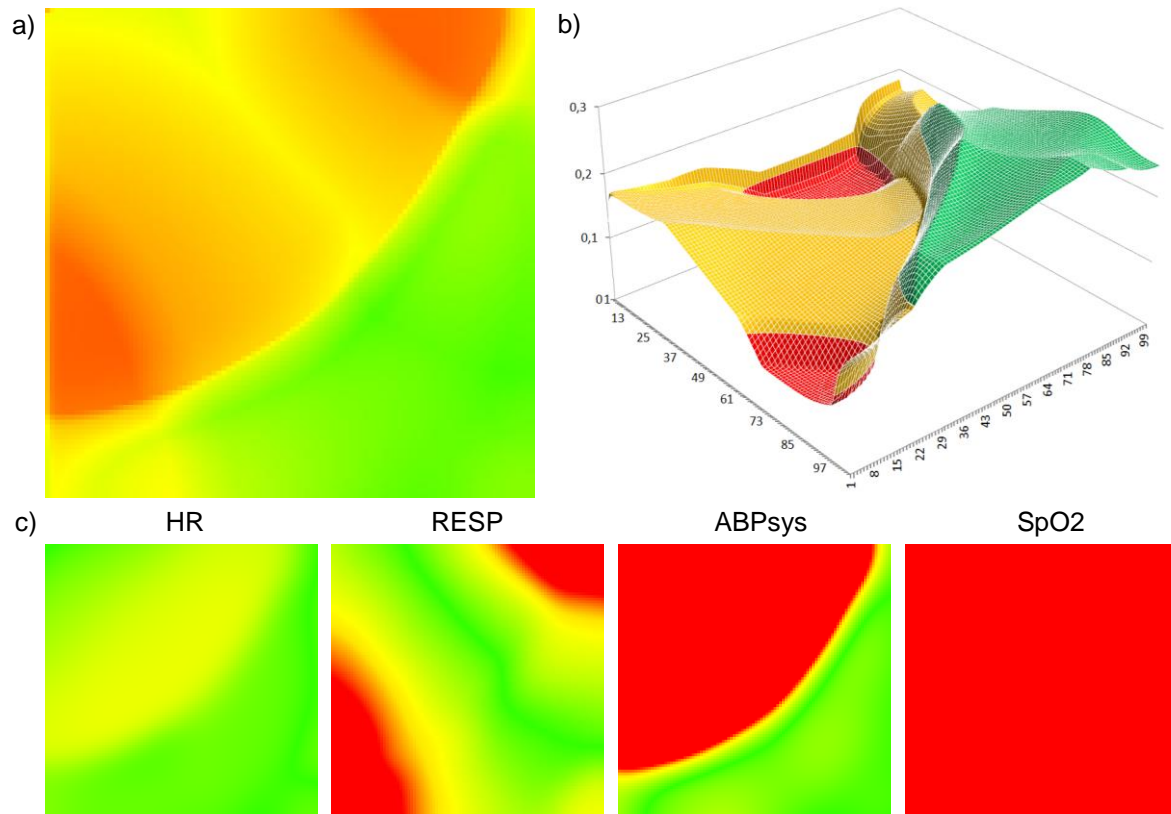
Figure 7.39 a) The single plane RAG map, b) the clustered 3D map, and c) the component planes view for each vital sign for the Self-Organizing Map formed after 10 epochs of the full DS1 dataset.

a)

b)

c)

| HR | RESP | ABPsys | SpO2 |
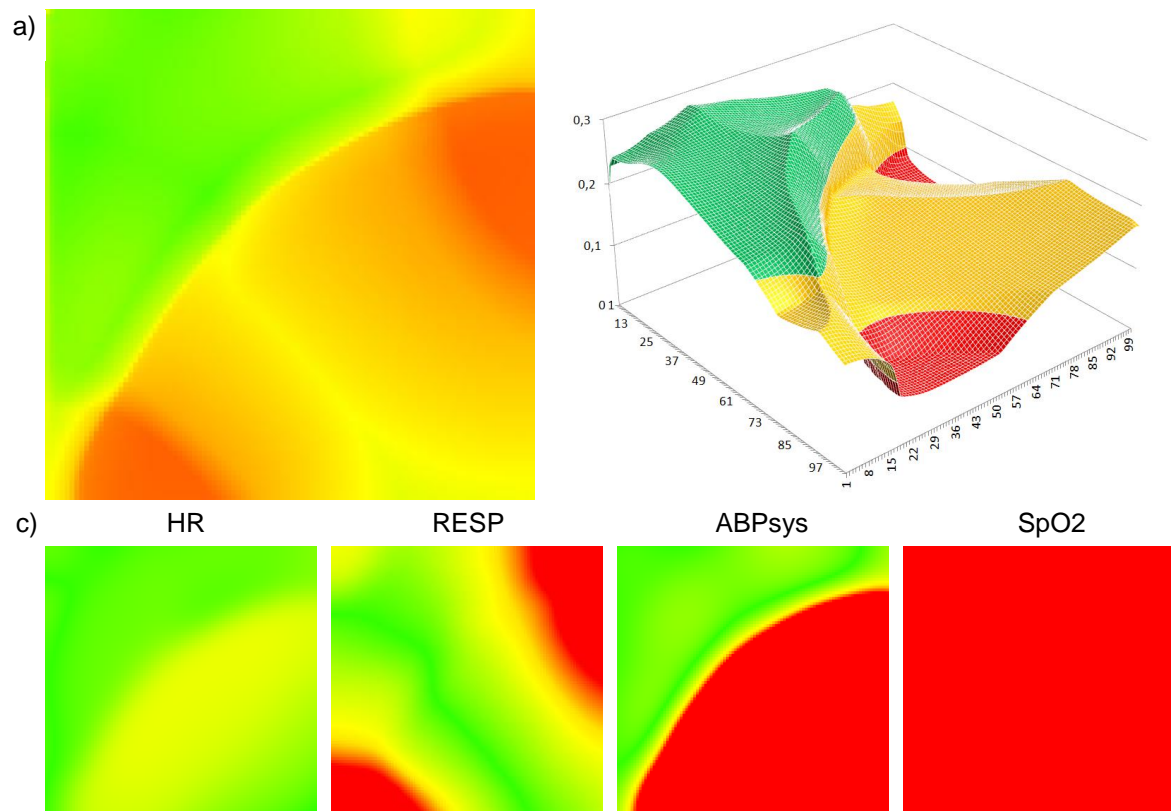
Figure 7.40 a) The single plane RAG map, b) the clustered 3D map, and c) the component planes view for each vital sign for the Self-Organizing Map formed after 20 epochs of the full DS1 dataset.

241

By comparing the single plane representation for all three training runs, it can be observed that the organization of the map was successful in case of trials with 10 and 20 epochs, where all variables (represented by RAG colours) are smoothly varying on the map. For this reason the trial with 5 epochs can be considered as unsuccessful. By comparing the map weights distribution for 10 and 20 epochs in figure 7.39b and 7.40b it can be noticed that by the 10th epoch the map converges (orders) to closely resemble its final state. Between the 10th and 20th epoch neuron weights spread out relatively evenly over the input space weights vectors while retaining their topological order found during the ordering phase.

The analysis of the component planes for all three training runs, indicated that 5 iterations over the DS1 data set were not enough for the SOM algorithm to organize the weights for the ABPsys dimensions. This, in turn, was successful with 10 and 20 epochs. Moreover, further analyses of component maps reviled that in all three cases the SOM algorithm was unsuccessful to cluster the SpO2 dimension. In order to establish what have caused such results, the investigation focused on the data set and its distribution. It turned out that SpO2 measurements for almost 40% of all types of samples in the DS1 data set are incomplete (the SpO2 measurements for those records are set to zero). Because during the normalization process the minimum and maximum values for the normal SpO2 signal were set to 94-100%, the zero value for the SpO2 in such settings has been normalized to much larger negative values. Larger negative values had a stronger effect on the training processes, which causes the SOM algorithm to bias to zero for this dimension of the input weights vector. As a result often missing SpO2 parameter has no influence on the SOM clustering process. Hence, it was consider as a redundant variable in this experiment and its value during the SOM testing process was set to zero for all the samples in the DS2 dataset.

In order to evaluate the classification performance of the of the proposed algorithm, it was necessary to define first when the alarm should be issued as well as when to consider it as a "real alarm" and as a "false alarm". Although the MIMIC database is accompanied by the patient status alarms (file extension .al), which refer to events that (as determined by the ICU monitor) require medical intervention (for example, observation of heart rate or blood pressure outside of pre-set limits) as well as monitoring condition alarms (file extension .in), which refer to events that interfere with the function of the monitor (for example, transducer malfunction or signal saturation), it should be noted that these alarms are as reported by the ICU monitors, including all false alarms. Therefore, in order to distinguish real from false alarms an additional NEWS-based rule which looks at correlations between individual parameters (and their alarm thresholds) has been proposed in this research to evaluate the SOM algorithm and consequently compare its results with the alarms generated by the NEWS scoring system applied to those three parameters. Individual parameter alarms used for annotation here refer to an individual parameter reaching the NEWS score of 3. According to this rule, only alarms which were triggered in one of the following cases are accepted as real alarms:

- The first and more severe alarm case states that minimum two out of three monitored parameters must generate the alarm (i.e. [HR=3 & RESP=3] or [HR=3 & ABPsys=3] or [RESP=3 & ABPsys=3]),

- The second alarm case is less severe and tends to identify correlations between multiple vital signs that lead to the potential alarm, when one parameter score 3 and two other parameters score 2 in NEWS scale. This approach follows the rule that no vital signs based life threatening events can occur in separation i.e. when there is a health risk most of the vital signs will go out of their normal ranges.

Finally to provide a fair comparison of the proposed algorithm and the NEWS scoring system, physiologically impossible values such as zero value for any two of the vital signs are excluded and such sample is being marked as incomplete. It is accepted however when one value in the sample is missing what resemble real life situations when due to patient movements, monitor conditions/noise or imperfections in monitoring scheme etc. some values produced by the vital signs monitors may lead to false alarms.

For such annotated DS2 dataset the quantitative analysis of the algorithm performance measure the number of true-positives, true-negatives, false-positives and false-negatives for each test sample. These were defined as follows:

8) true-positive (TP), defined as the number of samples that represent real alarms, which have been correctly classified as the high risk health status (correctly identified);

9) true-negative (TN), defined as defined as the number of samples that represent normal health status, which have been correctly classified as the low or medium risk health status (correctly rejected);

10) false-positive (FP), defined as the number of samples that represent normal health status but were incorrectly labelled as high risk health status (incorrectly identified);

11) and false-negative (FN), defined as the number of samples that represent real alarms, which have been incorrectly classified as the low or medium risk health status (incorrectly rejected).

In order to compare the performance of this QRS detection algorithm with other proposed methods, the accuracy (Acc), sensitivity (Se), specificity (Sp) and positive predictive (+P) values were calculated using the equations (88), (89), (90) and (41) respectively. Table 7.14 summarizes the test results obtained on the DS2 dataset. Looking at individual records the detection accuracy vary from 98.00% to 99.95%, sensitivity vary from 44.44% to 100%, specificity vary from 98.16% to 99.97%, and positive predictive value vary from 0.51% to 90.26% largely depending on the number of alarm and incomplete samples. The worst performance measure for the proposed algorithm is the positive predictive value (also called precision), which vary considerably from record to record. The lowest observed positive predictive value in records 241 and 293 are due to a very small number of alarm samples that were present in both records. A comparable small proportion of number of alarm samples to number of all samples in the record is also present in records 253 and 271.

| DS2 | | | | SOM Risk Classification | | | Classification Test | | | | Statistical Measures | | | |
| Record | Total Samples | Alarm | Normal | Incomplete | Alarm | Normal | | TP | FP | FN | TN | Acc (%) | Se (%) | Sp (%) | +P (%) |
					High	Medium	Low								
055	131,250	129	127,134	3,987	212	31,392	99,177	72	140	57	130,514	99.85	55.81	99.89	33.96
211	75,991	742	59,335	15,914	681	70,849	4,461	523	158	219	75,091	99.50	70.49	99.79	76.80
213	171,093	90	162,389	8,614	111	22,513	148,469	61	50	29	170,953	99.95	67.78	99.97	54.95
230	66,796	305	66,411	80	250	17,026	49,518	206	44	99	66,447	99.79	67.54	99.93	82.40
232	92,799	181	77,520	15,098	396	63,600	28,803	133	263	48	92,355	99.66	73.48	99.72	33.59
240	107,226	491	36,073	70,662	2,284	98,443	6,499	316	1,968	175	104,767	98.00	64.36	98.16	13.84
241	70,163	9	67,869	2,285	94	22,543	47,524	4	90	5	70,064	99.86	44.44	99.87	4.26
248	120,116	52	93,600	26,464	283	58,856	60,977	42	241	10	119,823	99.79	80.77	99.80	14.84
253	149,414	32	148,473	909	140	18,425	130,849	25	115	7	149,267	99.92	78.13	99.92	17.86
254	149,415	363	144,629	4,423	295	108,483	40,635	196	99	167	148,952	99.82	53.99	99.93	66.44
259	165,224	823	163,850	551	741	80,238	84,245	515	226	308	164,175	99.68	62.58	99.86	69.50
271	110,884	118	105,279	5,487	409	60,059	50,414	71	338	47	110,428	99.65	60.17	99.69	17.36
293	176,367	7	133,201	43,159	788	129,158	46,419	4	784	3	175,574	99.55	57.14	99.56	0.51
403	149,999	151	149,343	505	238	66,431	83,328	117	121	34	149,727	99.90	77.48	99.92	49.16
409	152,343	2,805	143,307	6,231	2,183	36,280	113,880	1,804	379	1,001	149,159	99.09	64.31	99.75	82.64
415	147,655	67	69,631	77,957	255	102,100	45,298	67	188	0	147,400	99.87	100.00	99.87	26.27
449	149,433	1,047	143,623	4,763	1,002	145,054	3,367	699	303	348	148,083	99.56	66.76	99.80	69.76
466	241,406	2,658	227,672	11,076	2,812	213,387	25,207	2,022	790	636	237,958	99.41	76.07	99.67	71.91
467	144,930	5,067	132,801	7,062	4,970	73,453	66,507	4,486	484	581	139,380	99.27	88.53	99.65	90.26
Total	2,572,504	15,137	2,252,140	305,227	18,144	1,418,300	1,135,577	11,363	6,781	3,774	2,550,116	99.59	75.07	99.73	62.63

Table 7.14 Results of performance evaluation for the proposed SOM classification algorithm using DS2 dataset from MIMIC Database.

Adding to this a relatively large number (when compared to the total number of samples) of incomplete samples (also present in records 240, 248, 293 and 415) that were marked as alarms (false positives), results in lower than an average positive predictive value for all those records. This indicates that many of the positive results in those records are false positives. Thus it will be necessary to follow up any positive result with a more reliable test or human intervention in order to obtain a more accurate assessment as to whether the life threatening even has occurred or not. Nevertheless, results obtained using the SOM algorithm compare favourably with those obtained with threshold based NEWS model, and shows superior performance in identifying potential health risk and reducing number of false alarms for incomplete samples, as evidenced in Section 8.3.3 where these two methods are compared.

Chapter

8

Discussion of Results and Conclusions

8.1   Introduction

A variety of wireless personal vital signs monitors, both for medical and fitness purposes, are either already in the market, or under development at prototype stage. Chapter 2 outlined some notable wearable systems developed in recent years along with a brief description of their applications. A vast number of those projects focused on the on-body sensing technologies aiming at new sensing, actuating, communicating and processing different physiological parameters. Whereas sensors and actuators are essential to promote SWS adoption amongst the population, as some researchers like Chan et al. (2012) claim, to further boost SWS's adoption and release their commercial value the new data processing as well as new integration and interoperability paradigms are needed. Both framework and processing model presented in this work falls squarely within the remit of this claim. Hence, this chapter attempts to explain how both, the framework and data processing model, presented in this work, constitute the future of SWS's, by drawing upon published results to evaluate them in light of applications and other methods.

8.2   Framework architecture evaluation

The portable vital signs monitoring framework proposed in Chapter 4 along with the Mobile Inference Engine presented in Chapter 6 is a typical example of distributed middleware architecture, characterised by high-level abstraction to simply construction of distributed systems. This infrastructure provides a distributed environment for developing application-level components at sensor-, smartphone- and server-level. Each of these applications relies on platform-specific middleware to manage its life cycle and execution, with some of them providing off-the-shelf services such as data processing, communication, storage or security. Consequently, the application and the middleware are tightly coupled, and thus the middleware design to plays a crucial role in achieving the quality attribute requirements of the system.

As a result, middleware creates new challenges and issues for software architecture evaluation methods. Their poor design, implementation errors or lack of any feature, may eventually lead to the failure of applications to meet their requirements, especially when

considering middleware horizontal nature of providing a wide range of mechanism for many vertical application domains. Moreover, middleware 1) are more dependent upon the mechanisms and services provided by the infrastructure, 2) are typically more complex and 3) the access to the adequate range of stakeholders with clear business goals for the middleware systems is usually not possible. Therefore, standard existing software evaluation methods, which are typically driven by the specific goal of an application, fail to successfully evaluate middleware architectures (Liu, Gorton, Bass *et al.* 2006).

Approaches to middleware architectures evaluation that exists in literature (Dobrica and Niemela 2002, Mattsson, Grahn and Mårtensson 2006), can be considered as pertaining to two streams: software architecture (SA) evaluation and middleware evaluation. The SA evaluation methods, as formulated by Babar and Gorton (2004) focus on understanding the relationship between software architecture and one or more quality attributes, to ensure that the system ultimately achieves its goals while still supporting its functional requirements. SA evaluation methods can be divided into four main categories: experience-based which rely on experience and domain knowledge of developers or consultants (Avritzer and Weyuker 1999); simulation-based, which rely on a high level implementation of some or all of the components in the software architecture (Aquilani, Balsamo and Inverardi 2001); mathematical modelling based on mathematical proofs and methods for evaluating mainly operational quality requirements such as performance and reliability (Reussner, Schmidt and Poernomo 2003); and scenario-based that tries to evaluate a particular quality attribute by creating a scenario profile that forces a very concrete description of the quality requirement such as maintainability, reliability, flexibility and so on (Mattsson, Grahn and Mårtensson 2006). As discussed and evaluated by Liu et al. (2006), the scenario-based evaluation methods best fit the purpose of middleware architectures evaluation in terms of their inputs, the roles involved and the output of the methods. Examples of scenario based methods include: Software Architecture Analyses Method (SAAM) (Kazman, Abowd, Bass *et al.* 1996), Architecture Trade-Off Analyses Method (ATAM) (Kazman, Klein, Barbacci *et al.* 1998), and Architecture Level Modifiability Analyses (ALMA) (Bengtsson, Lassing, Bosch *et al.* 2004).

The second stream of methods for the middleware architectures evaluations covers evaluation of components and middleware as opposed to the whole architecture of the system. The most notable example of such evaluation methodologies is i-Mate (Liu and Gorton 2003) which is similar to the first phase of ATAM, and requires stakeholders to input the business requirements for the middleware to be acquired. The evaluation of performance and scalability using i-Mate is conducted in a lab environment by running a predefined benchmark application on all the candidate middleware with the rest of the evaluation test environment remaining identical. Other methods and techniques available, like those presented in (Kanoun, Kaaniche and Laprie 1997, Cecchet, Marguerite and Zwaenepoel 2002) are less complex and focus rather on specific quality attributes evaluation. Moreover some quantitative quality attributes, such as performance and availability (Dong, Kumar, Duvur *et al.* 2004) can be accessed

through measurement, analytical modelling and simulations. Other identified middleware evaluation practices can be considered as a unsystematic process of random middleware features' evaluation. All these methods, however do not fully embrace the relationship between a middleware technology and the application in which the middleware is considered as the features and constraints of the implementation of the application. Therefore they do not explicitly address all the issues, discussed above that concerns middleware evaluation (Liu, Gorton, Bass *et al.* 2006).

8.2.1 MEMS: A Method for Evaluating Middleware Architectures

The evaluation of middleware architectures of the proposed portable vital signs monitoring framework, is done according to a Method for Evaluating Middleware Architectures (MEMS), proposed by Liu et al. (2006). MEMS, like SAAM and ATAM is a scenario-based evaluation method dedicated to middleware architectures, which evaluate its multiple quality attributes, by leveraging generic qualitative and quantitative evaluation techniques such as prototyping, testing, rating, and analysis. It measures middleware architectures by rating multiple quality attributes with respect to key scenarios that describe the behaviour of the middleware in particular context.

MEMS presents a structured approach, which addresses the challenges of middleware architectures evaluation discussed in the previous section. MEMS is different from i-Mate as it is concerned with evaluating alternative solutions using a single middleware infrastructure. The business goals are imposed on the output of MEMS and are not a portion of the method. The evaluation process is driven by concerns about the quality attributes for specific designs using the middleware. MEMS can also be used to enhance the development process by emphasizing quality attributes and focusing on architectural design decisions in projects like the one presented in this document, where the middleware is evolving rapidly to support emerging technologies, and agile application development methods. In such cases MEMS can help to uncover potential problems at early stage, making it cheaper and quicker to fix design problems.

MEMS evaluation process can be divided into two stages made of seven steps as depicted in Figure 8.1 The first stage is the development of the evaluation plan. This stage encompasses the first four steps of MEMS as defined in (Liu, Gorton, Bass *et al.* 2006):

1. **Determine critical quality attributes** – Unlike SAAM, ATAM or i-Mate, MEMS is not driven by quality requirements derived from the business goal. Instead it addresses general quality attributes such as performance, scalability, modifiability or Configurability. These critical quality attributes should be nominated by middleware architect who have architectural knowledge of the system. The main concerns for the quality attributes must be specified in this step. The purpose of this is to set a context for next step which generates key scenarios for the quality attributes.

2. **Develop key scenarios** – In this step real scenarios are created for each quality attribute or for its associated sub-concerns. This should be done in a descriptive form with precise quality attribute requirements, as quality attributes by themselves are too abstract for analysis. These scenarios should be ranked according to the priority of quality attributes.

3. **Define metrics for each quality attribute** – This step defines metrics for both quantitative as well as qualitative attributes. Quantitative attributes use measurements or simulation techniques. For qualitative attributes the weighted scoring method to consolidate results is proposed. It requires a definition of consistent rating criteria with clear and unambiguous rating scales.

4. **Identify key architecture patterns for the proposed scenarios** – This step define the middleware architecture possible for the implementation of the given scenario. Middleware normally provides multiple mechanism and services to support same functionality. The solution proposed in this step will be used for implementation and evaluation in the second stage of MEMS.

The second stage is the actual evaluation process which involves the last three steps:

5. **Develop prototype applications for the key scenario** – In this step a prototype implementation is produced.

6. **Evaluate quality attributes through experiments or experts ratings** – This step evaluates individual quality attributes. For quantitative attributes the evaluation focuses on producing the metrics values such as throughput for performance. For qualitative attributes evaluators give ratings against the rating scales defined for each quality attribute.

7. **Analyse measurements and present the evaluation results** – The evaluation results are visually presented in a way that clearly identifies the rating with regards to individual attributes. The result of this step could further be used to analyse trade off or refine middleware design if the agile development process is adopted.
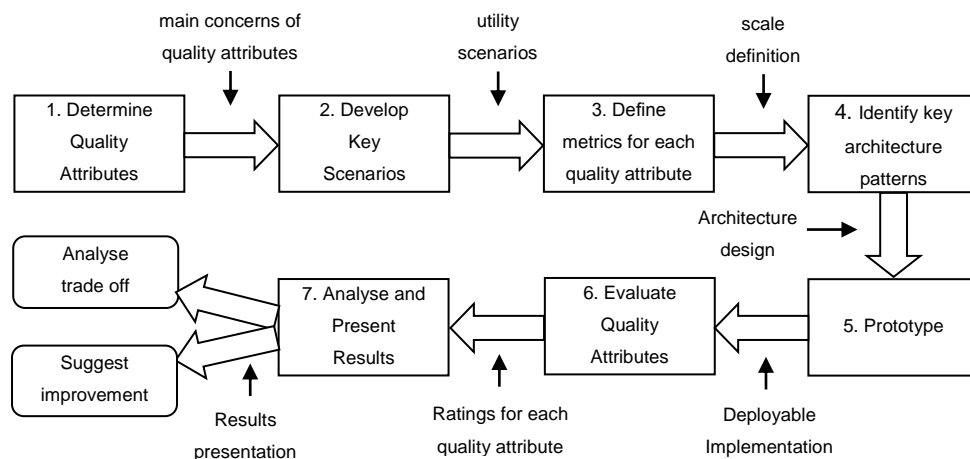
Figure 8.1 MEMS steps adopted from (Liu, Gorton, Bass *et al.* 2006)

8.2.2 Sensor middleware evaluation

In order to evaluate the effectiveness of the proposed sensor middleware in enabling a cost effective, flexible applications design and implementation, the following quality attributes have been defined, according to MEMS model. These are:

1) Scalability – which concerns the ability of the node middleware to support the growth of the application through increase in number of sensors, increase of sensor related processing capabilities, sensor maintenance and reconfiguration.

2) Heterogeneity – which concerns the ability of the node to provide interface to meet/response to various types of hardware and network conditions, such as application specific network protocols.

3) Extensibility – which concerns the ability to extend the system. Unlike scalability, which focus on availability of mechanism to support growth of the application at runtime such as by adding/removing new sensors or processes, the extensibility focus on the level of effort, such as implementation of the configuration file or deployment tasks, required to implement likely extensions. Extension can be through the addition of new functionality, new resources, new conditions, or through modification of current functionality.

4) Programmability – which concerns the ability of the middleware to reduce the programming effort by simplifying or hiding the complex implementation of services form the developers who build applications on top of the middleware. This is the trade-off between flexibility and simplicity of developing the application, where development of the application can be done either by simply configuring the descriptor file (high simplicity) or through customised implementation in the code (high flexibility).

Following such defined quality attributes and their metrics a sensor middleware evaluation scenario have been proposed, where simple AccelECG application source code, available from TinyOS contributed code repository, was compared in terms of programmability quality attribute and memory requirements with the source code of the same application, but implemented with the used of the sensor node middleware, proposed in this work. Given the general use case scenario, for the sensor node, which include sampling, pre-processing and transmitting the data samples to the personal server, the AccelECG application, that implements all these functionalities represents a valid test scenario. To further evaluate the remaining quality attributes of interest, both direct TinyOS as well as middleware based AccelECG implementations have been extended by adding two additional custom sensors: a temperature probe (AccelECG_TEMP application) and piezoelectric respiratory sensor (AccelECG _TEMP_RESP application). The processes and efforts associated with development / extension of both implementations have been compared and assessed.

Quality Attribute	Scale definition		
	High	**Medium**	**Low**
Scalability	No manual changes in configuration are required to grow the application size by adding data source/ sensor/ sensor node or service and any of its functions.(m)	Changes in configuration or implementation with templates, patterns or frameworks are required to add data source/ sensor/ sensor node or service and any of its functions.(s)	Custom implementation with APIs is required to add data source/ sensor/ sensor node or service and any of its relevant functions.
Heterogeneity	Only changes in configuration files or implementation for setting properties are required to integrate new hardware or network protocol.	Changes in configuration and implementation with templates, patterns or frameworks are required to integrate new hardware or network protocol. (s), (m)	Custom implementation with APIs is required to integrate hardware or network protocol.
Extensibility	Only changes in configuration or implementation for setting properties are required.	Changes in configuration and implementation with templates, patterns or frameworks are required. (s), (m)	Custom implementation with APIs is required to extend application functionality.
Programmability	Support for visual application development and custom implementation with scripts and configuration files (m)	Support for application development with configuration files and custom implementation with templates, patterns or frameworks. (s)	Support for out-of-box application development by using APIs.

Table 8.1 Quality attributes' rating scale definition.

The quality attributes assessment, made during the implementation of the AccelECG, AccelECG_TEMP and AccelECG _TEMP_RESP applications, showed that applications' development process facilitated by the proposed sensor node middleware, can be flexible, but also less effort, and what follows, less cost intensive than direct TinyOS implementations. This is due to the following quality attributes:

1) Scalability (Rank: Medium) – The middleware improves the scalability of the application in case when new sensor must be deployed, enabling to reuse existing code. It can be achieved with minimal modifications to the internal structure of the existing application. The whole process is limited to adding a new Sensor configuration component and connecting it with the main application in the configuration file. Depending on the application, existing or new Sensor component operation, can be further extended by the Processor component. The Processor component takes buffer of sampled values, obtained form Sense interface, and performs a set of filtering and data conversion activities on it. In than return those values that are of interest as data streams. If required a new or existing Filter component can be contacted to the Processor

component to further extend its functionality without modifying the internal structure of the Processor. In turn, the addition of a new sensor or data processing/filtering capabilities to the direct TinyOS implementation of AccelECG, required an extensive integration into an internal structure of the application, what led to a complete redesign of its all components and configuration files. Therefore, the scalability attribute of the application implemented with the middleware ensured high flexibility in implementation of any custom processing or filtering functionality, as well as high simplicity in adding new sensors through changes in configuration.

2) Heterogeneity (Rank: Medium) – The middleware enables the network protocols heterogeneity, which can be simply achieved by adding or modifying the NodeProtocol's Operation component. The Operation components are responsible for performing an externally customizable set of operations in response to the input from the network interface or call from the application main module. Using Dispatcher pattern of NodeProtocol component, a developer is able to invoke operations using a parameterised interface, based on data identifier. The dispatcher is independent of what commands the application handles, or what processing those handlers perform. Adding a new network operation with the middleware requires a single wire to NodeProtocol component, what in case of direct Tiny OS implementations is equivalent to extensive interference into the internal structure of the application and redesign of network packets and network commands. Because the network protocol extension of the AccelECG application required both changes in the configuration files and some custom implementation of the Operation components, therefore the heterogeneity attribute of the middleware has be ranked as medium. Future works will need to ensure that the network protocol customizations are possible through configuration files only.

3) Extensibility (Rank: Medium) – The comparison of direct TinyOS implementations with middleware based implementations revealed that the second approach, required only minimal modifications to the internal structure of the old applications (configuration files) in order to extend it to work with other sensors or network protocols. On the other hand the simplicity of the application extension is supported by the ability to integrate (also in a structured way) custom components and processing logic, what preserves the flexibility of the middleware. However, in the case of the evaluation scenario, most of the other extensions could have been achieved by adding new components or setting components' properties, what in case of direct Tiny OS implementation, again required extensive interference with the internal structure of the application.

4) Programmability (Rank: Medium) - Direct TinyOS implementation is much more complex than middleware based development. Developers implementing directly on TinyOS stack must manually wire all subsequent components, implement their interfaces, control application network usage, parsing, buffering data etc.. In turn, by

using the middleware a developer focus on the application specific processing related to the actual data of interest, with all configuration being done a priory by the model. Moreover, the data flow in the direct TinyOS application it is not evident, due to split phase operation, what greatly complicate the maintenance and debugging. The proposed middleware alleviates this problem with the Processor components. Processor components are single, data stream related processing units, which can be easily debugged and analysed. Because in its current form the proposed middleware does not come with the complementary visual editor, which would allow visually assemble an application, its programmability quality attribute has been ranked as medium. The medium rank is due to its ability to considerably limit the complexity of the programming problem through, templates and patterns, when compared to direct TinyOS implementation by using APIs.

Additionally to qualitative attributes, quantitative measures have been used to compare the complexity of direct and middleware based implementations of AccelECG, AccelECG_TEMP and AccelECG _TEMP_RESP applications. These comparison, presented in Figure 8.2 includes: the number of lines of code, number of explicit application events and their memory requirements. In term of the number of lines of code, this has been decreased by almost 50% with the use of middleware. Moreover, by using the middleware the number of explicit application events that the developer must control, in order to simply get sensor readings, decreased to 4 events what is independent from the number of sensors. These events include: connection made, command received, sensing done and connection closed. However, such a simplification comes at the price of a slight increase in the size of the binary code deployed on the mode (ROM) as well as heap size (RAM). This, however, remains well within the limits of program flash memory size of the commercially available sensor platforms (e.g. 48KB ROM + 10KB RAM for SHIMMER).

Application	Lines of code		Explicit events		ROM [KB]		RAM [B]	
	Direct	Middl.	Direct	Middl.	Direct	Middl.	Direct	Middl.
AccelECG	334	201	8	4	8.96	10.87	576	1074
AccelECG_TEMP	357	212	10	4	9.26	10.91	634	1086
AccelECG _TEMP_RESP	402	231	12	4	9.87	10.96	690	1094
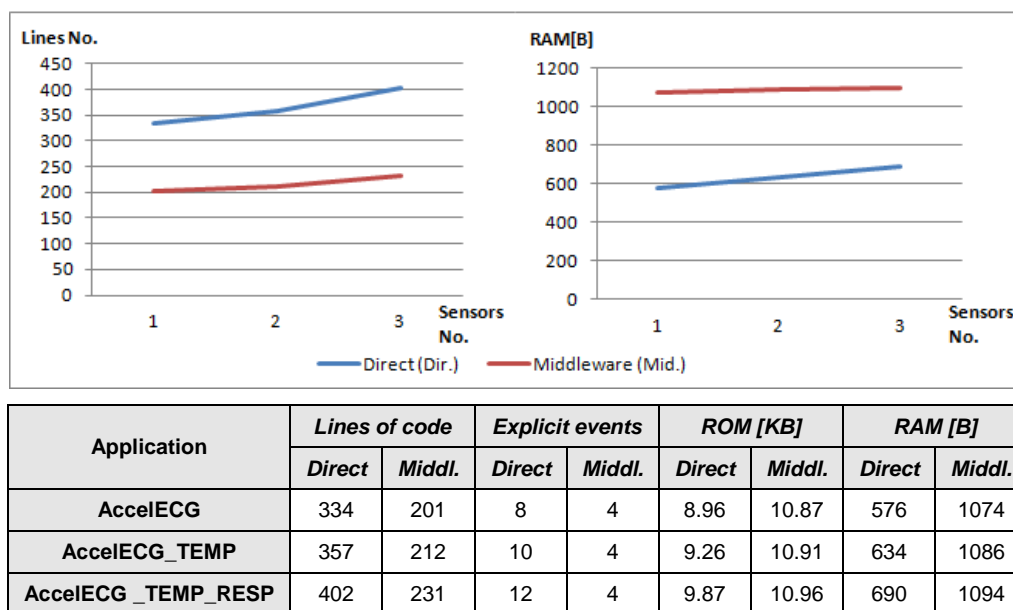
Figure 8.2 Comparison of middleware based applications with their direct TinyOS implementations.

8.2.3 Smartphone middleware evaluation

Based on the middleware architecture presented in section 4.4, a prototype of the vital signs monitoring system presented in shapter 5 has been developed using both sensor and personal server middleware. The system is capable of continuously monitor and analyse five vital signs using two sensor nodes. The first sensor node is a chest strap presented in Section 5.2, capable to measure ECG, temperature, and respiratory rate, which use off-the-shelf SHIMMER wireless sensor platform with number of sensors such as SHIMMER ECG daughter card, NTC type thermistor and piezoelectric sensor. It has been programmed using our TinyOS sensor node middleware. In turn, the second sensor node used in our case study is a commercially available wireless pulse oximeter, which was rapidly integrated with the aid of the personal server framework. The prototype of the personal server has been implemented for CLDC 1.1 and MIDP 2.0 profiles in Java programming language. The prototype system was tested on phoneME Java Virtual Machine (Phoneme n.d.) that can run on Windows, IOS and Android.

In order to evaluate the effectiveness of the proposed smartphone based middleware for data aggregation and processing, in enabling a cost effective, flexible applications design and implementation, four quality attributes defined in the previous section have been evaluated. These are:, Heterogeneity, Scalability, Programmability and Code/applications reusability. The evaluation of the personal server implementation conducted during the aforementioned application design process revelled some important model inherent properties for all four quality attributes as follows:

1. Scalability (Rank: High) – In order to ensure that networks and application are scalable the middleware proposed the abstraction of networks, nodes and sensors which impose particular WBAN organizational structure presented in Figure 8.3.
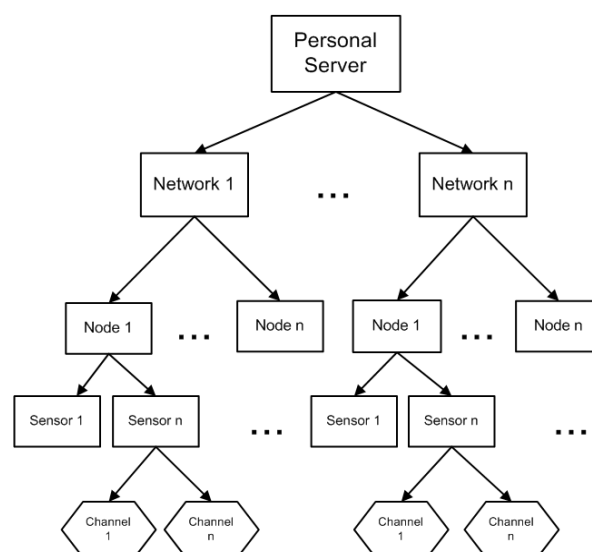
Figure 8.3 WBAN organizational structure

Using such WBAN organisation we can add or remove sensor nodes automatically, at the application settings level, by concurrently searching on multiple wireless networks, which represent implemented and registered communication stacks. When the sensor node is discovered it can be registered with the application. Each network can have multiple sensor nodes, which are responsible for data processing, and radio communication. Each node might work in three modes: as a network forwarder only, as a data acquisition node only or both at the same time. If a node is working in either of the two last modes it must have at least one sensor deployed. Each sensor, in turn, has at least one channel responsible for data sampling at the ADC input. This organisational structure presents a versatile, robust and transparent solution to wireless data acquisition with varying number of sensor nodes and network topologies.

2. Heterogeneity (Rank: Medium) – One of the main requirements for the Personal Server middleware is to ensure ease integration of multiple third party sensor devices and other network communication protocols. This was obtained through the sensor abstraction model, which simplifies data acquisition and network coordination by hiding the whole network specific functionality, what enables to perceive sensors as a data publishing service within any type of network.

The integration of a Bluetooth wireless communication protocol with the presented framework has been easily achieved with the use of ComStack interface, which implements protocol specific commands and further registers itself with NetworkManager. Implementation of custom CommStack required a discoverDevices() method, which places the device into devices discover inquiry mode and searches for devices in range. Those devices that responded to the inquiry are returned as a vector of NodeInfo objects to the application. As discovery inquiry is executed in a separate thread the cancelDiscovery() method is required to stop the inquiry. In turn, search Services() method, searches for services on a remote device and return the URL addresses, which are used to establish the connection. Finally a method getName() to return the name of the communication stack has to be implemented in order to allow dynamically register the communication stack with the framework.

In turn, the integration of AccelECG_TEMP_TESP sensor node has been achieved with the use of Node interface and NodeInfo abstract class, which were implemented to meet these node protocol requirements. Detailed description of the porting process with the Bluetooth SHIMMER example can be found in section 4.4.4.1.2. Below is the summary of all object's method signatures that have been implemented:

```
public abstract class NodeInfo extends Persistent {
    public NodeInfo();
    public abstract Node getNode();}
```

255

```
public interface Node extends Persistent {
    public NodeConnection getConnection();
    public abstract void getFrame();
    public abstract NodeInfo getNodeInfo();
    public abstract String[] getSensors();
    public boolean isAvailable();}
```

3. Programmability (Rank: High) – This is enabled by the high-level abstraction of application design process using embedded inference engine and its rapid deployment model facilitated by the visual data flow composition using visual editor. This is facilitated through complementary visual Inference Engine Editor, that comes with the package. It enables AI specialist along with medical experts to build data processing models by interlacing together different components and controlling the application logic with scripts. The editor allows the instant deployment of such models remotely on patient mobile devices. This approach shifts focus from software and hardware development, to medical and health process implementation.

4. Extensibility (Rank: Medium) – Extensions to the core functionality of the middleware and applications are enabled through components' templates and framework design patterns. During the implementation of the prototype, many new functions and algorithms have been implemented including: new QRS detection algorithm, SVM and TASOM model, as well as new input and output processing components such as data normalization components or SOM visualisation.

   Finally, in order to verify that the advantages previously identified do not negatively affect the system performance, the evaluation was extended beyond the development model to also look at how the network overheads and inference engine model affect the sample execution time. Figure 8.4 shows significant correlation between execution time and the complexity of the inference algorithm (measured by the Self-Organizing Map dimension). However, it also shows that the network overheads associated with the increase of number of sensor data streams does not affect the execution time, which proves the quality of the presented model.
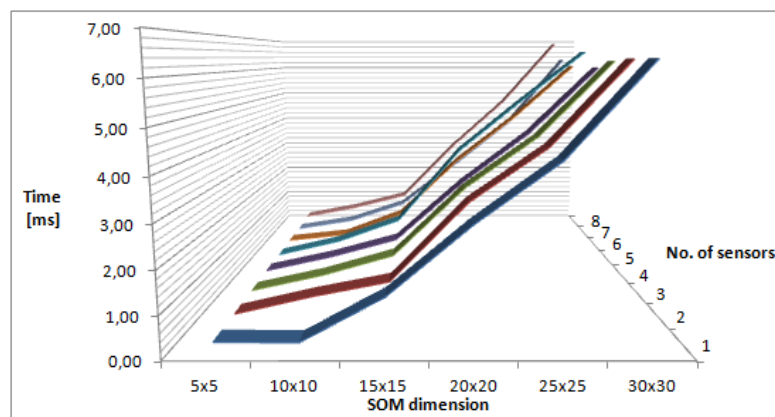
Figure 8.4 Impact of number of sensors and complexity of inference algorithms on sample execution time

8.3　Model Evaluation

The model for patient specific well-being analysis, presented in chapter 7, is an example of a typical machine learning application. An essential part of new machine learning methods validation, is the statistical evaluation of experimental results. The evaluation starts with the implicit hypothesis, that the proposed method/enhancement yields an improved performance over the existing algorithm(s). To accept or reject the hypothesis a number of test data sets were selected for testing. The proposed algorithm were run and the quality of the resulting models was evaluated by calculating the number of True Positives (TP), True Negatives (TN), False Positive (FP) and False Negatives (FN) on each data set. In order to ensure comparability of proposed method with other algorithms, an appropriate performance measure(s) were used, such as: classification accuracy, sensitivity, specificity, positive predictive value, negative predictive value or failed detection. Both numerical and statistical measures were defined according to evaluation standards proposed in the literature. These are discussed for each algorithm individually, due to different interpretation of results for multiple data sets.

However, comparison of numerical and statistical measures of different algorithms in isolation lacks an important information: it does not tell us whether the difference between results is random or not. To address this in the evaluation, a paired t-test was computed, which checks whether the average difference in performance of each pair of algorithms over the data set is significantly different from zero. T-test assumes that each sample is drawn from a data set of normal or near-normal population distribution.

The t test requires first to state a null and an alternative hypothesis in such a way, that they are mutually exclusive, that is, if one is true, the other must be false; and vice versa. The hypotheses concern a new variable d, which is based on the difference between performance scores of two classifiers x_i and x_j on a given data set, defined as:

$$d = x_i - x_j \tag{109}$$

For such formulated test variable the test is conducted, which aims to reject a null hypothesis, and therefore confirm alternative hypothesis. For each test a null and alternative hypothesis is formulated, each making a statement about how the true difference in performance scores μ_d between algorithms is related to some hypothesised value D. The D value in this case is set to zero what can be interpreted as no performance difference between algorithms. The implicit hypothesis which states that the proposed method/enhancements yields an improved performance over the existing algorithm(s) lead to one-tailed test, since an extreme value on only one side of the sampling distribution would cause to reject the null hypothesis. A paired t-test looks at the difference between paired values in two samples, takes into account the variation of values within each sample, and produces a single number known as a t-test statistic defined as:

$$t_0 = \frac{(\overline{x}_i - \overline{x}_j) - D}{SE} = \frac{\overline{d}_{ij} - D}{S_{ij}/\sqrt{n}}$$

where $\overline{x}_i$ is the mean performance score of first algorithm, $\overline{x}_j$ is the mean of performance score of second algorithm, d is the mean difference between scores, D is the hypothesised difference between population means, SE is the standard error, and S_{ij} is the standard deviation of the differences.

If the findings are unlikely, given the null hypothesis, the null hypothesis is rejected. This involves comparing the one-tailed P-value for a given t value, to the significance level. The P-value is found from the t distribution with n-1 degrees of freedom. The significance level used in this study is set at the level of 5% (0.05). Depending how the hypothesis was formulated for the test the null hypothesis get rejected either when the P-value is less than the significance level or when the 1-P-value is less than the significance level.

Following this methodology in the reminder of this section three proposed algorithms are evaluated and their results are discussed in context of data sets they were trained and tested on. This is considered to fulfil the requirement of testing the second overall-project hypothesis.

8.3.1 QRS detection algorithm evaluation

The comparative study on the performance of the QRS detection algorithms, presented in Section 7.5, in terms of the number of false negative and false positive detections is shown in Table 8.2. It provides a quick overview of the achieved detection results for the selected most noisy records. The results were ordered by the total number of false detections, with the most accurate algorithm on the right hand side and least accurate algorithm on the left hand side.

In this study we considered only those reported results which are fully comparable. That means that those evaluated on all 48 records of the MIT-BIH arrhythmia database, which is the

Rec. no.	All Beats	Number of false-negative (FN) detections & Se (%)						Number of false-positive (FP) detections & +P (%)					
		Alg. 1	Alg. 2	Alg. 3	Alg. 4	This work	Alg. 5	Alg. 1	Alg. 2	Alg. 3	Alg. 4	This work	Alg. 5
104	2229	2	12	7	1	6	0	2	63	3	7	5	14
105	2572	31	6	22	19	17	8	73	28	53	7	21	18
106	2027	2	17	2	20	0	0	7	2	1	21	4	2
108	1763	19	28	47	2	10	4	207	55	50	10	8	12
113	1795	0	1	1	11	1	0	0	0	2	10	2	3
116	2412	22	20	25	27	21	16	4	3	4	4	2	8
121	1863	7	1	0	0	1	0	5	6	1	13	2	2
200	2601	7	0	2	9	4	0	6	47	3	4	5	6
203	2980	41	53	61	7	22	11	57	27	14	3	18	5
208	2955	21	30	19	10	14	8	5	5	9	3	4	5
209	3005	2	0	2	9	0	0	4	1	2	2	7	0
210	2650	10	29	41	5	2	6	3	8	2	16	7	3
221	2427	1	4	1	8	0	0	3	0	1	4	0	0
222	2483	92	0	37	0	3	0	112	3	40	1	8	0
223	2605	2	1	2	22	1	1	2	0	0	4	0	0
228	2053	9	6	6	2	5	6	26	76	19	10	9	7
232	1780	5	0	0	2	0	0	7	20	3	14	10	18
233	3079	3	7	3	8	1	2	0	0	0	7	0	0
Total	43279	276	215	278	162	108	62	523	344	207	140	112	103
		99.36%	99.50%	99.36%	99.63%	99.75%	99.86%	98.80%	99.21%	99.52%	99.68%	99.74%	99.76%

Table 8.2 Comparison of the numbers of false-positives (FPs) and false-negatives (FNs) for most noise records of the MIT-BIH arrhythmia database.

most common standard for assessing the performance of QRS detectors. Only these works comply with the requirements of comparability and reproducibility discussed by Kohler et al. (2002) and can be compared in an objective manner. In case where only parts of a standard database have been used, the intention of the authors usually was to show the performance of the algorithm on particularly difficult records, such as records with pathological or very noisy signals. However such reported results are not truly reliable, as the algorithm may have been tuned to perform perfectly on such pathological signals but not on a normal ECG.

Following these criteria, four competitive methods which report the highest accuracy have been selected from the literature, along with the (Alg. 1) algorithm proposed by Paoletti and Marchesi (2006) where the curve-length concept was first applied to QRS detection. These four algorithms are: (Alg. 2) Adnane et al's (Adnane, Jiang and Choi 2009) method based on morphological characteristics; (Alg. 3) Hamilton's and Tompkins's (1986) method based on digital filters and first derivative; (Alg. 4) Zhang and Lang's (2009) method based on multi-scale mathematical morphology filtering; as well as the most accurate (Alg.5) method of Manikandan and Soman (2012) based on Shannon energy envelope (SEE) estimator and Hilbert transform.

By looking at the sensitivity (Se), the positive predictive value (+P) and the failed detection (FD), the proposed algorithm was classified as the second most accurate algorithm, just after (Alg. 5). With respect to computational load, algorithms (Alg. 1,2, and 4) as well as the one proposed in this work, can be classified as of low computational load with applicability to wearable devices, while methods used in (Alg. 3 and 5) will have a medium to high computational load according to the classification proposed by Kohler (2002).

In order to investigate how the introduction of pre- and post-processing stages together with the new peak detection logic affected the performance of the curve-length, the results obtained with the proposed algorithm were compared with results reported in (Paoletti and Marchesi 2006). As it can be observed in Table 8.2, the proposed algorithm significantly reduced the number of false negatives from 276 to 108. This improved the sensitivity of the algorithm by 0.39% and had an effect on algorithm performance for almost all noisy records (105, 108, 203, 208, 210, 222 and 228). The only exception to this is record 116 where the number of false negatives did not change. When looking at the false positives, the improvement in positive predictive value with our proposed algorithm is even more significant, resulting in a 0.94% difference. The total number of false positives reduced from 523 to 112 and has been observed with all noise records, even with record 116, which characterise by smaller QRS complexes than in the other records.

In order to test whether the difference between results of the proposed classifier and results of its five competitive methods, obtained over MIT-BIH database records, is non-random; a paired t-test was formulated. The aim of the test is to evaluate if the presented algorithm performs statistically better than other compared algorithms, therefore the test hypothesis is formulated as follows:

Null hypothesis $H_0 : \mu_d = 0$ *Presented algorithm performs no better (same or higher average FD) on average Failed Detection (FD) then other compared algorithm(s).*

Alternative hypothesis $H_1 : \mu_d > 0$ *Presented algorithm performs better (lower average FD) on average Failed Detection(FD) then other compared algorithm(s).*

Such formulated test does not tell us however whether two algorithm, for which null hypothesis is accepted, is performing statistically better or maybe the same as the proposed algorithm. Therefore, in order to eliminate this ambiguity the second test was formulated which aim is to establish if the presented algorithm performs statistically worse than other compared algorithms. Therefore algorithm(s) which fail the first and the second test are those that perform statistically the same as the proposed algorithm. Hence, the second test hypothesis is formulated as follows:

Null hypothesis $H_0 : \mu_d = 0$ *Presented algorithm performs no worse (same or better average FD) on average Failed Detection (FD) then other compared algorithm(s).*

Alternative hypothesis $H_1 : \mu_d < 0$ *Presented algorithm performs worse (higher average FD) on average Failed Detection(FD) then other compared algorithm(s).*

For the first test the null hypothesis get rejected when the P-value is less than the significance level. For the second test the null hypothesis rejection criteria is when the 1-P is less than the significance level. Results of this evaluation are presented in Table 8.3.

Rec. no.	Number of false-positive (FP) + Number of false-negatives (FN)						$d =$ Alg. n – This work				
	Alg. 1	Alg. 2	Alg. 3	Alg. 4	This work	Alg. 5	Alg. 1	Alg. 2	Alg. 3	Alg. 4	Alg. 5
104	4	75	10	8	11	14	-7	64	-1	-3	3
105	104	34	75	26	38	26	66	-4	37	-12	-12
106	9	19	3	41	4	2	5	15	-1	37	-2
108	226	83	97	12	18	16	208	65	79	-6	-2
113	0	1	3	21	3	3	-3	-2	0	18	0
116	26	23	29	31	23	24	3	0	6	8	1
121	12	7	1	13	3	2	9	4	-2	10	-1
200	13	47	5	13	9	6	4	38	-4	4	-3
203	98	80	75	10	40	16	58	40	35	-30	-24
208	26	35	28	13	18	13	8	17	10	-5	-5
209	6	1	4	11	7	0	-1	-6	-3	4	-7
210	13	37	43	21	9	9	4	28	34	12	0
221	4	4	2	12	0	0	4	4	2	12	0
222	204	3	77	1	11	0	193	-8	66	-10	-11
223	4	1	2	26	1	1	3	0	1	25	0
228	35	82	25	12	14	13	21	68	11	-2	-1
232	12	20	3	16	10	18	2	10	-7	6	8
233	3	7	3	15	1	2	2	6	2	18	1
Total	799	559	485	302	220	165	-	-	-	-	-
Mean ($\bar{d}$)							32.17	18.83	14.72	4.78	-3.06
S. Dev. (s)							64.33	25.66	25.20	15.28	7.03
S. Error (SE)							15.16	6.05	5.94	3.60	1.66
t_0							2.12	3.11	2.48	1.33	-1.84
Test 1: P_1							0.0244	0.0032	0.0120	0.1011	0.9586
Test 2: P_2							0.9756	0.9968	0.9880	0.8989	0.0414

Table 8.3 Hypothesis testing using t-test and difference in number of failed-detected (FD) beats for all compared algorithms tested on most noise records of the MIT-BIH arrhythmia database.

The results of the first test indicated that the proposed method is statistically significant and outperforms algorithms (1), (2) and (3) at significance level of 5% (p=0.0244, p=0.0032, p=0.0120 respectively). On the other hand significance against algorithms (4) and (5) was inconclusive (p=0.1011, p=0.9586). These results have led to the second test for which the null hypothesis was rejected in case of algorithm (5) (p=0.0414) only, which therefore can be considered as the method which offers statistically better performance than the proposed algorithm. In turn, the second test was inconclusive for algorithms (1), (2), (3) and (4) (p=0.9756, p=0.9968, p=0.880, p=0.8989 respectively). Therefore the proposed algorithm offers statistically better performance than algorithms (1), (2) and (3), same performance as algorithm (4) and statistically worse performance than algorithm (5), characterised by high computational load.

Overall the proposed detector shows good performance for signal with noise, even in the presence of pronounced muscular noise baseline artefacts, or smaller QRS complexes. The salient characteristics of this algorithm are its simplicity and low computational resources requirements. The algorithm does not require prior training or knowledge of the frequency spectrum of the ECG signal. Moreover it does not require any predefined amplitude thresholds, as the decision is based on the condition, which is satisfied or not, by signal dependent functions of mean and standard deviation. As the observed test results compare favourably with published results for other QRS detectors, the proposed algorithm becomes a very attractive option for application on wearable devices with limited storage, short battery life and a relatively low computational power.

8.3.2 Binary ECG classification algorithm evaluation

The evaluation of the proposed binary ECG classification algorithm with other methods, unlike QRS or SOM model evaluations, focus on the comparison of their overall statistical performance measures on DS2 dataset, rather on differences in results for individual records. Such evaluation methodology for this algorithm was selected because of three reasons. Firstly, during the literature review it has been found out that the type of classification to normal or abnormal beats, adopted in the proposed algorithm, is very rare, and the access to results of other classifiers for such stated problem is limited. Secondly, although in most cases the published results are for the MIT-BIH Arrhythmia database records, different authors use different inter-patient dataset divisions between training/validation and testing dataset, what further limited the comparability between results. Finally, most of the published results are for the multiclass ECG beat classifiers which do not include classification results for individual records, but rather focus on confusion matrices, which summarise the allocations to each class for the whole test dataset.

Consequently, the comparison of the proposed algorithm was based on the results contained in confusion matrices for those methods, which consistently used the same database

and inter-patient dataset division for testing, according to the data division proposed initially by de Chazal et al. (De Chazal, O'dwyer and Reilly 2004). The proposed method was compared with multiclass classification algorithms, which results have been decomposed to binary problems by mapping their multiclass confusion matrices to binary confusion matrices as illustrated in Figure 8.5. However, it should be noted that some of the algorithms being compared may have been tuned to perform detection of a specific abnormality and therefore their results may not fully represent capabilities of the classifier used in those methods.

Figure 8.5 Multiclass confusion matrix to binary confusion matrix decomposition.

Following these criteria, five competitive methods which used the same DS1 and DS2 data division have been selected from the literature, along with the algorithm proposed by De Chazal et al. (De Chazal, O'dwyer and Reilly 2004) where the DS1 and DS2 data division was first applied to ECG abnormalities detection. These five algorithms are: Jiang's & Kong's (Wei and Kong 2007) block-based neural network (BbNNs) classifier with Hermite transform coefficients and the time interval between two neighbouring R-peaks; Huang's et al. (Huang, Liu, Zhu *et al.* 2014) method based on random projections and Support Vector Machine (SVM); Mar's et al. (Mar, Zaunseder, Marti *et al.* 2011) Multilayer Perceptron (MLP) model with morphology, WT and interval features; Zhang's et al. (Zhang, Dong, Luo *et al.* 2014) SVM ensemble with morphology and interval feature; as well as Llamedo's and Martinez's (Llamedo and Martinez 2011) Linear Discriminant Analysis (LDA) method with Wavelet Transform feature extraction. Table 8.4 compares the proposed algorithm's results with those of other methods. The results were ordered by the accuracy, with the most accurate algorithm at the top of the list.

Method	Features	Classifier	TP	FP	FN	TN	Acc (%)	Se (%)	Sp (%)	+P (%)
Jiang et al. (2007)	Hermite coefficients, intervals	BbNN	6,125	533	1,639	41,303	95.62	78.89	98.73	91.99
Huang et al. (2014)	Random projections, intervals	SVM	4,966	2327	487	41,931	94.34	91.07	94.74	68.09
Mar et al. (2011)	Morphology, WT, intervals	MLP	5,075	4,564	351	39,497	90.07	93.53	89.64	52.65
Zhang et al. (2014)	Morphology, intervals	SVM	5,148	4,869	405	39,142	89.36	92.71	88.94	51.39
De Chazal et al. (2004)	Morphology, intervals	LDA	5,126	5,814	327	38,444	87.65	94.00	86.86	46.86
Llamedo et al. (2011)	WT, intervals	LDA	5,260	9,918	181	34,270	79.65	96.67	77.55	34.66
This method	WT, intervals	SVM	4,616	695	825	43,536	96.94	84.84	98.43	86.91

Table 8.4 Classification performance comparison of the proposed method with other methods on DS2 records of the MIT-BIH arrhythmia database.

By looking at the Accuracy (Acc), the proposed algorithm with its 96.94% was be classified as best out of all compared methods. With respect to specificity (Sp) and the positive predictive value (+P) the proposed algorithm with its 98.43% and 86.91% respectively, was classified as the second best, just after method proposed by Jiang and Kong (Jiang and Kong 2007). High specificity and positive predictive value in this case means that an algorithm returned substantially more relevant beats with abnormalities than irrelevant normal beats indicating the higher probability that in case of a positive test, the beat really is the abnormal beat. However, on a downside of the proposed method is its relatively low proportion of actual positives, which are correctly identified as such, when compared to other methods. The sensitivity (Se) of 84.84% for the proposed algorithm was outperformed by all compared methods apart of the method proposed by Jiang and Kong (Jiang and Kong 2007).

In order to assess the statistical significance of the proposed diagnostic test, the Sensitivity and Specificity confidence intervals at the 95% confidence level have been calculated according to and compared. The comparison of confidence intervals for sensitivity and specificity for the proposed algorithm did not overlap with the confidence intervals of other methods, what indicates that results obtained with the proposed algorithms are statistically significant at the level of 5%. Results of this evaluation are presented in Figure 8.6

Sensitivity (Se) + 95% CI

	Jiang	Huang	Mar	Zhang	De Chazal	Llamedo	This method
– Upper End	79.79%	91.81%	94.17%	93.38%	94.62%	97.13%	85.78%
– Lower End	77.96%	90.28%	92.84%	91.99%	93.34%	96.16%	83.86%
– Se	78.89%	91.07%	93.53%	92.71%	94.00%	96.67%	84.84%

Specificity (Sp) + 95% CI

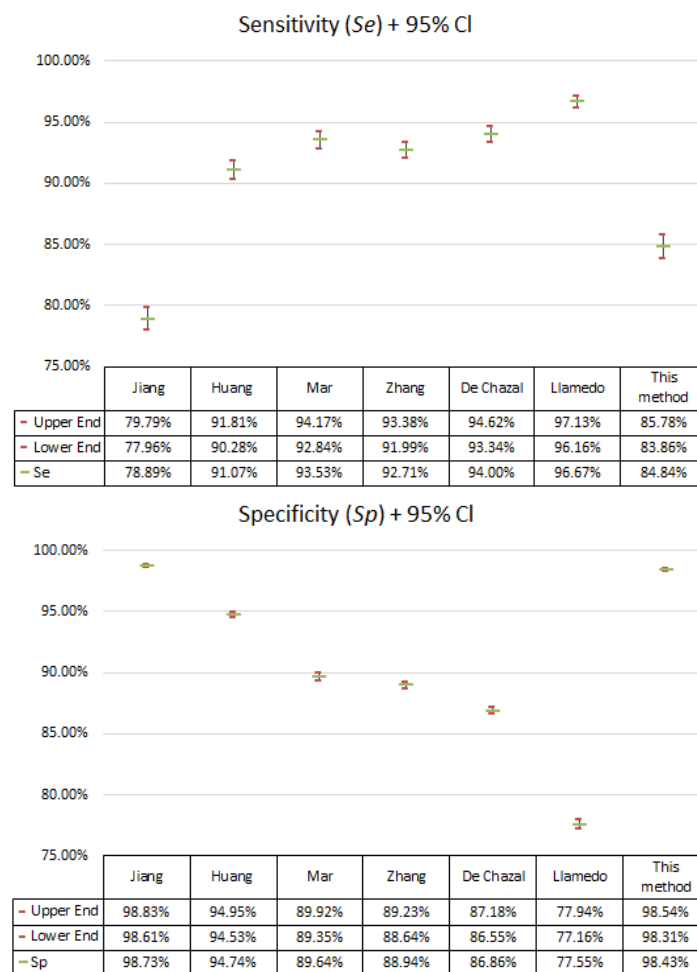|	Jiang	Huang	Mar	Zhang	De Chazal	Llamedo	This method
– Upper End	98.83%	94.95%	89.92%	89.23%	87.18%	77.94%	98.54%
– Lower End	98.61%	94.53%	89.35%	88.64%	86.55%	77.16%	98.31%
– Sp	98.73%	94.74%	89.64%	88.94%	86.86%	77.55%	98.43%

Figure 8.6 The sensitivity (Se) and specificity (Sp) values with 95% confidence intervals for all compared methods.

8.3.3 SOM health map evaluation

The comparative study of the performance of the SOM algorithm presented in Section 7.7 in contrast to traditional threshold based NEWS model used in the hospital wards in UK, is shown in Table 8.5. It provides an overview of the achieved detection results with focus on the Sensitivity and Positive predictive value on DS2 dataset for both methods.

Rec. no.	Inc. samp.	NEWS							SOM						
		TP	FN	Se (%)	incl. Inc.		excl. Inc.		TP	FN	Se (%)	incl. Inc.		excl. Inc.	
					FP	+P (%)	FP	+P (%)				FP	+P (%)	FP	+P (%)
055	3,987	125	4	96.90	4,213	2.88	290	30.12	72	57	55.81	140	33.96	140	33.96
211	15,914	579	163	78.03	414	58.31	30	95.07	523	219	70.49	158	76.80	84	86.16
213	8,614	55	35	61.11	311	15.03	39	58.51	61	29	67.78	50	54.95	18	77.22
230	80	9	296	2.95	7	56.25	0	100.00	206	99	67.54	44	82.40	41	83.40
232	15,098	29	152	16.02	573	4.82	0	100.00	133	48	73.48	263	33.59	104	56.12
240	70,662	20	471	4.07	4,464	0.45	55	26.67	316	175	64.36	1,968	13.84	1,799	14.94
241	2,285	0	9	0.00	7	0.00	0	100.00	4	5	44.44	90	4.26	70	5.41
248	26,464	16	36	30.77	1,484	1.07	61	20.78	42	10	80.77	241	14.84	83	33.60
253	909	0	32	0.00	29	0.00	7	0.00	25	7	78.13	115	17.86	71	26.04
254	4,423	200	163	55.10	799	20.02	456	30.49	196	167	53.99	99	66.44	60	76.56
259	551	15	808	1.82	127	10.56	42	26.32	515	308	62.58	226	69.50	221	69.97
271	5,487	32	86	27.12	4,846	0.66	10	76.19	71	47	60.17	338	17.36	269	20.88
293	43,159	4	3	57.14	11,680	0.03	3667	0.11	4	3	57.14	784	0.51	216	1.82
403	505	0	151	0.00	94	0.00	0	100.00	117	34	77.48	121	49.16	42	73.58
409	6,231	144	2,661	5.13	5,663	2.48	396	26.67	1,804	1,001	64.31	379	82.64	292	86.07
415	77,957	0	67	0.00	15,215	0.00	46	0.00	67	0	100.00	188	26.27	12	84.81
449	4,763	12	1,035	1.15	399	2.92	52	18.75	699	348	66.76	303	69.76	286	70.96
466	11,076	4	2,654	0.15	1,156	0.34	531	0.75	2,022	636	76.07	790	71.91	786	72.01
467	7,062	5,066	1	99.98	4,964	50.51	0	100.00	4,486	581	88.53	484	90.26	160	96.56
Total	305,227	6,310	8,827	41.69	56,445	10.05	5,682	52.62	11,363	3,774	75.07	6,781	62.63	4,754	70.50

Table 8.5 Comparison of the numbers of true-positives (TPs), false-negatives (FNs) and false-positives (FPs) (including and excluding incomplete samples) for DS2 dataset records of the MIMIC database.

By looking at number of TP, FN and FP as well as their associated sensitivity (Se) and the positive predictive (+P) values, the proposed algorithm clearly outperforms the NEWS model in all those categories. Compared to NEWS model, the number of TP and FN for the SOM algorithm, respectively, increased from 6,310 to 11,262 and decreased from 8,827 to 3,774. This resulted in the increase of Sensitivity by 33.38% from 41.69% for NEWS model to 75.07% for the SOM model. The most substantial difference in performance of both algorithms can be observed however with respect to number of FP (and associated +P value) for the data set which included the incomplete samples. For this category of assessment, the SOM model improved the positive predictive value by 52.58%, from 10.05% for NEWS model, to 52.63% for SOM model, by decreasing the number of false-positive detections from 56,445 to 6,781. Further confirmation of a superior performance of the SOM over NEWS model in reducing number of false alarms, comes from comparison of both models in terms of number of FP (and associated +P value) on the data set which excluded the incomplete samples. For this category of assessment the SOM model improved the positive predictive value by 17.88% from 52.62% for NEWS model to 70.50% for SOM model and decreased the number of false-positives from 5,682 to4,754.

In order to test whether these differences between results of the proposed classifier and those of the NEWS model, are non-random; a paired t-test was formulated. The aim of the test is to evaluate if the presented algorithm performs better than currently used threshold based technique in terms of identifying potential health risk and reducing number of false alarms for

incomplete samples. Therefore the test hypothesis, in terms of number of FN, FP$_{incl}$, and FP$_{excl}$ respectively, was formulated as follows.

Null hypothesis $H_0 : \mu_d = 0$ *Presented algorithm performs no better (same or higher average FN / FP$_{incl}$ / FP$_{excl}$) on average than NEWS model.*

Alternative hypothesis $H_1 : \mu_d > 0$ *Presented algorithm performs better (lower average FN / FP$_{incl}$ / FP$_{excl}$) on average then NEWS model.*

For such formulated test the null hypothesis get rejected when the P-value is less than the significance level. Results of this evaluation are presented in Table 8.6.

Rec. no.	NEWS			SOM			d_x = NEWS – SOM		
	FN	FP$_{incl}$	FP$_{excl}$	FN	FP$_{incl}$	FP$_{excl}$	x = FN	x = FP$_{incl}$	x = FP$_{excl}$
055	4	4,213	290	57	140	140	-53	4,073	150
211	163	414	30	219	158	84	-56	256	-54
213	35	311	39	29	50	18	6	261	21
230	296	7	0	99	44	41	197	-37	-41
232	152	573	0	48	263	104	104	-310	-104
240	471	4,464	55	175	1,968	1,799	296	2,496	-1,744
241	9	7	0	5	90	70	4	-83	-70
248	36	1,484	61	10	241	83	26	1,243	-22
253	32	29	7	7	115	71	25	-86	-64
254	163	799	456	167	99	60	-4	700	396
259	808	127	42	308	226	221	500	-99	-179
271	86	4,846	10	47	338	269	39	4,508	-259
293	3	11,680	3,667	3	784	216	0	10,896	3,451
403	151	94	0	34	121	42	117	-27	-42
409	2,661	5,663	396	1,001	379	292	1,660	5,284	104
415	67	15,215	46	0	188	12	67	15,027	34
449	1,035	399	52	348	303	286	687	96	-234
466	2,654	1,156	531	636	790	786	2,018	366	-255
467	1	4,964	0	581	484	160	-580	4,480	-160
Total	8,827	56,445	5,682	3,774	6,781	4,754	-	-	-
						Mean ($\bar{d}$)	265.95	2613.89	48.84
						S. Dev. (s)	610.41	4138.46	924.02
						S. Error (SE)	140.04	949.43	211.98
						t_0	1.90	2.75	0.23
						Test: P	0.0369	0.0065	0.4102

Table 8.6 Hypothesis testing using t-test and difference in number of false-negatives (FN), false-positives including incomplete samples (FN$_{incl}$), and false-positives excluding incomplete samples (FN$_{excl}$), between NEWS and SOM techniques, tested on DS1 records of the MIMIC database.

The results of the test indicated that the proposed SOM method is statistically significant and outperforms NEWS model in terms of number of false-negatives and false positives for the data set which included the incomplete samples at significance level of 5% (p=0.0369, p=0.0065 respectively). On the other hand significance test in terms of number of false positives for the data set which excluded the incomplete samples was inconclusive as the probability of p=0.4102 did not allow to reject the null hypothesis for this test at the level of 5%.

These results have led to the conclusion that the proposed SOM based health monitoring performs significantly better than currently used NEWS threshold based technique, in terms of identifying potential health risk and reducing number of false alarms for records characterised by large number of incomplete samples.

8.4   Experimental Results

In order to validate the accuracy of the proposed system as well as estimate its cost, the prototype system has been built and tested in lab on volunteering adult test subjects Validation tests for the proposed experimental hardware design was conducted in lab on group of healthy volunteers. The validation process focused on signal accuracy and filtering. During each test four primary vital signals: ECG, Respiratory Rate, Temperature and Pulse Oximetry, were collected using prototyped sensor devices, which design was presented in chapter 5. These results were then compared with reference signal obtained from the same test subjects using medically approved sensor devices used in ambulatory settings.

Results from the validation tests for the prototype system are presented in sections 8.4.1-8.4.4 below, followed by the rough guideline of the cost of the final commercial product in section 8.4.2.

8.4.1   Electrocardiograph (ECG) signal validation

Off-the-shelf SHIMMER ECG daughter card, without any further digital signal processing generates the ECG signal as presented in figure 8.7. This raw, noisy signal, was obtained from the SHIMMER by transmitting it wirelessly over Bluetooth to the application deployed on the Android device, where the signal was aggregated and an ECG graph was plotted.

The signal presented in figure 8.7, collected from a test subject at rest, contains all kinds of ECG signal noise. It is mostly affected by the significant power lines noise that cause 55/65 Hz sinusoidal interference as well as by moderate high-frequency (above 80 Hz) muscle noise and low-frequency high-bandwidth baseline wander components due to perspiration or respiration noise. This is due to the fully resting position of the monitored subject, where body movements did not affected the electrode impedance. Both low and high band FIR filters, discussed in chapter 5, successfully removed excessive noise, producing clean ECG pulsatile signal as presented in Figure 8.8.
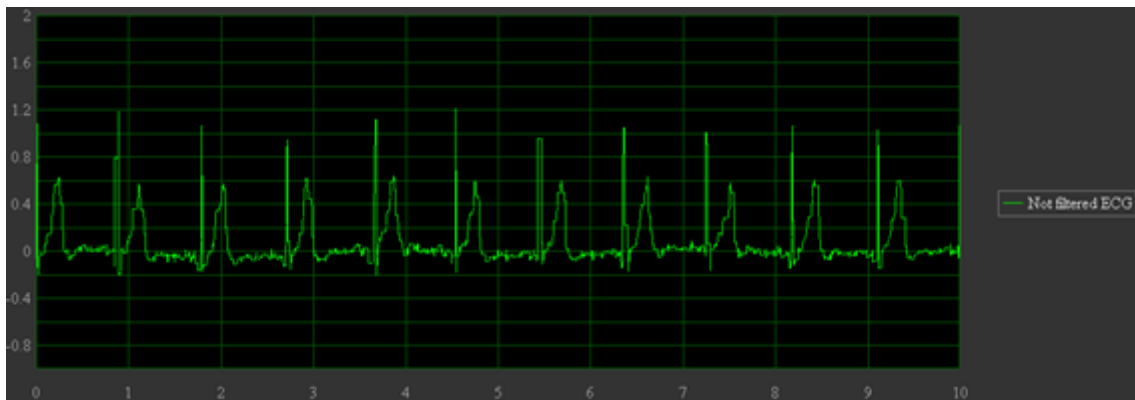
Figure 8.7 Raw, pulsatile ECG signals obtained by SHIMMER ECG module from a healthy test subject, using custom application deployed on the mobile device.

Figure 8.8 Filtered ECG signals obtained from SHIMMER, using FIR filters with custom application deployed on the mobile device.

A number of tests were carried out to validate the SHIMMER ECG daughterboard as a valid tool for acquiring ambulatory ECG (Mcgrath and Dishongh 2009). The tests consisted of validating the ECG amplifier and ADC performance by using calibrated input signals and a resting ECG from a healthy subject in normal sinus rhythm. Simulated ECG signals as well as an ECG recording from a healthy non-resting subject were used to validate the performance SHIMMER ECG daughterboard for use in ambulatory monitoring. Figure 8.9 shows a 1mV QRS amplitude (60 BPM) waveform generated by the Fluke MPS450 Patient Simulator (Fluke Biomedical 2010) captured by a SHIMMER ECG, while Figure 8.10 shows signal captured by MAC 3500 ECG Analysis System (Ge Healthcare 2010) for comparative signal quality purposes. Visual examination of the waveforms indicates that they compare well. The plotted waveform in Figure 8.9 and 8.10 was recognizable by a clinician as Normal Sinus Rhythm.

Figure 8.9 Simulated ECG signal of 1mV QRS amplitude captured by the SHIMMER ECG (Mcgrath and Dishongh 2009)

Figure 8.10 Simulated ECG signal of 1mV QRS amplitude captured by a MAC 3500 ECG Analysis System (Mcgrath and Dishongh 2009)

Additionally, as reported in (Mcgrath and Dishongh 2009) a 5.9 minute ECG recording containing 503 heart beats from a non-resting healthy subject during a moderate walk was captured by the SHIMMER ECG and also captured by a Medilog Holter (Medilog 2010) monitoring system. The R-R intervals and instantaneous heart rate (HR) identified by using the Medilog Holter ECG monitor software were compared against the R-R intervals calculated by using method proposed by Benitez et al. (Benitez, Gaydecki, Zaidi *et al.* 2001) to detect QRS segment using the Hilbert transform and R-R interval correction algorithms. Each automatically detected QRS point on the SHIMMER ECG was manually verified to ensure correct detection by using the QRS detection algorithm.

As reported the mean R-R interval for the SHIMMER acquired ECG was 0.7049 seconds, while the mean R-R interval for the Medilog acquired ECG was also 0.7049 seconds. The percentage difference between the R-R intervals for each acquisition was calculated on a beat-beat basis. The mean percentage error between the R-R intervals calculated, by using each acquisition, was found to be 0.0192 percent which can be considered negligible. These results indicate that the SHIMMER ECG in such configuration can be used to acquire ambulatory ECG from resting and non-resting human subjects for research application purposes. Moreover, SHIMMER ECG daughter card offers some advantages for ECG monitoring over bulky size and heavy ambulatory Holter monitors. The compact size allows short lead lengths to reduce noise. Also the SHIMMER ECG circuit uses an AC-coupled topology to improve signal quality when the subject is in motion. Periodic shifts in the zero-signal readout during periods of motion or lead-manipulation are normal and easily eliminated in post-processing.

8.4.2   Respiratory rate signal

As described in chapter 5 respiratory rate measurement method utilize piezoelectric sensor attached to the user's chest, which generate an electric potential in response to a

mechanical stress applied to an elastic material strap, which holds the sensor. According to the respiratory rate monitor block diagram, piezoelectric sensor is sufficient to give raw signals, which change with force applied to the sensor as presented in Figure 8.11.

a)
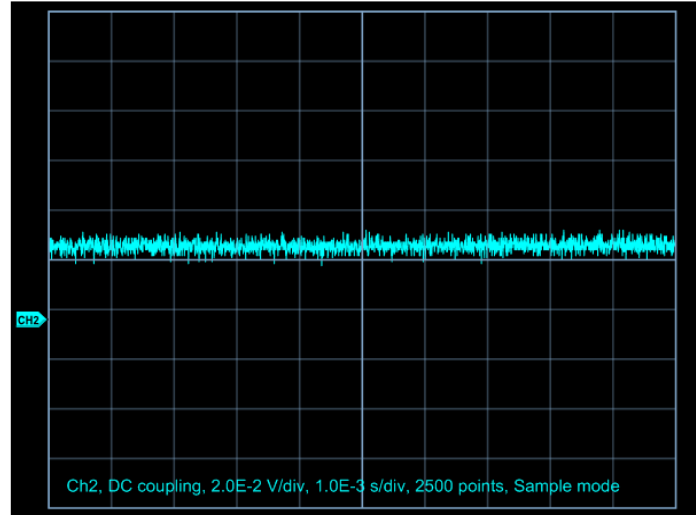
b)
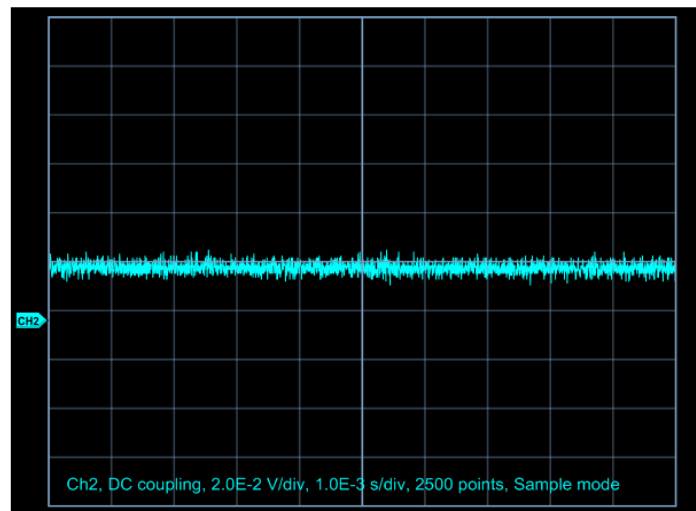
Figure 8.11 Signal obtained from oscilloscope using SleepSense piezoelectric sensor when a) breathing in and b) breathing out the air from the lungs (each division is 10 mV).

Such obtained signal however needs further processing. First of all, the signal is prone to be contaminated by noise caused by the body moves that are not related to breathing process. Additionally, due to the physiological nature of the signal, its amplitude is extremely small approximately between -1 to 1 mV for tidal volume exchange and between -5 to 10 mV for expiratory and inspiratory reserve volume exchange respectively.

The issues mentioned above were addressed with a low pass filter with cut-off frequency of 1 Hz to remove the motion noise and an amplifier to increase the AC signal component amplitude. Figure 8.12.presents the comparison of the initial raw piezoelectric signal with the filtered and amplified respiration signal obtained from the oscilloscope.
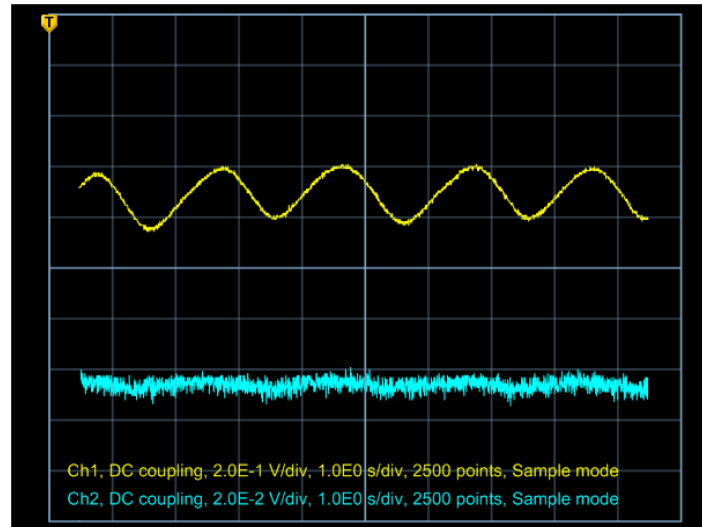
Figure 8.12 Filtered and amplified respiration signal obtained from oscilloscope using SleepSense sensor (each division is 500 mV).

Obtained signal was recognized by a clinician as a valid respiration signal and successfully compared with results obtained using RESPeRATE (Intercure n.d.), clinically proved portable device for respiratory and blood pressure monitoring, used in non-drug therapy of high blood pressure and the reduction of stress (Schein, Gavish, Herz *et al.* 2001).

As presented in Figure 8.12 the low pass filter accomplished it's tasks by removing the excess noise, present in the raw signal due to motion of the individual user, and passed only the signal that comes from the chest movements. The same filter also helped to attenuate noise at the 60 Hz frequency range, which is the noise from the power lines. The amplification stage has accomplishing its task amplifying the signal to the range of (2-3.5V). The amplification was achieved by using a non-inverting operational amplifier configuration with gain of 100 times. Amplified signal range corresponds to the initial potential range of (-5mV; 10mV), generated by the piezoelectric sensors, which has been shifted to (20mV; 35mV) using a fixed resistors supply voltage divider. Such obtained superimposed (AC) signal after amplification oscillates around 2.5V, what makes it possible to be sampled by the analog-to-digital converter.

8.4.3   Body temperature signal

Proposed thermocouple design incorporates MA100 thermistor which resistance varies with temperature. It measures the temperature representing it as voltage values. Each voltage value has a linear correspondence to one temperature value.

Both in-lab as well as human tests were conducted to test the proposed design's accuracy and response time. For this purposes the reference temperature probe, Braun PRT2000EU Age Precision Stick Digital Thermometer Braun Age Precision® Digital Stick Thermometer PRT2000 has been used. Table 8.7 presents how its temperature tolerance changes over 0°C to 50°C temperature range. Shaded area indicates the intended operating temperature range which corresponds to standard human body temperature ranges. The

thermal response time for the given MA100 thermistor in catheter assembly was tested by plunging the probe from reference temperature of 25°C (± 0.1°C) air to 5°C (± 0.1°C) air and still water. On average the response time was 15 sec and 2 sec respectively. These results show that MA100 thermistor used as a temperature probe maintains tight interchangeable tolerances with good thermal response time.

Temperature Range (°C)	Temperature tolerance (±°C)
0 to 20	0.2
20 to 35	0.15
35 to 39	0.1
39 to 42	0.15
42 to 45	0.15
45 to 50	0.2

Table 8.7 Temperature tolerance over 0°C to 50°C temperature range for MA100 thermistor.

8.4.4   Pulse oximetry signal

Proposed pulse oximeter design, presented in chapter 5, includes the LED circuitry, photodetector and current-to-voltage converter which can produce a raw, pulsatile signal. Figure 8.13 illustrates raw pulsatile signal obtained by placing infrared LED and the photodetector probe onto test subject's fingertip.

Ch1 DC coupling, 4.0E-2 V/div, 1.0E0 s/div, 2500 points, Sample mode
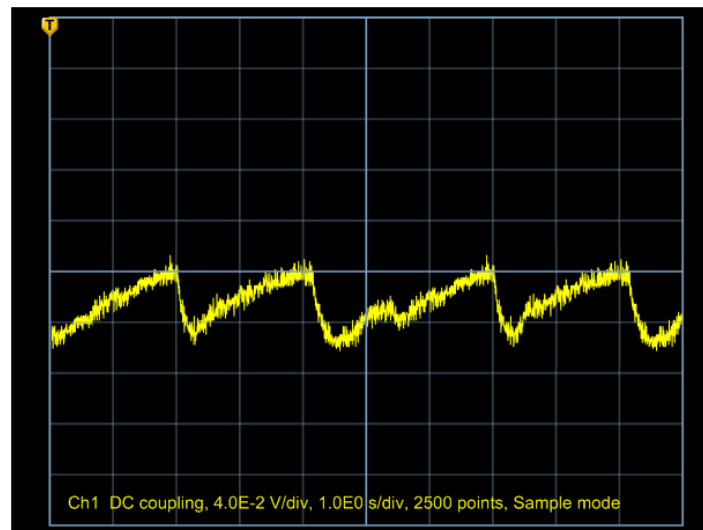
Figure 8.13 Raw, pulsatile signal obtained using IR LED.

Such obtained signal however presents some major drawbacks. Firstly, the pulsatile signal is superimposed on a much larger DC offset.. Secondly, the raw signal contains considerable amount of noise. This noise comes from many sources, along with the 60 Hz power line noise, motion noise from movements by the user, and noise from any surrounding light sources of a visible spectrum.

Due to these issues, the proposed design includes the post processing elements: a) a band pass filter to remove the excess noise and DC offset and b) an amplifier to increase the signal amplitude to the desired value. Figure 8.14 below presents the pulsatile signal for the infrared spectrum obtained from the same test subject using prototype pulse oximeter with post processing blocks.
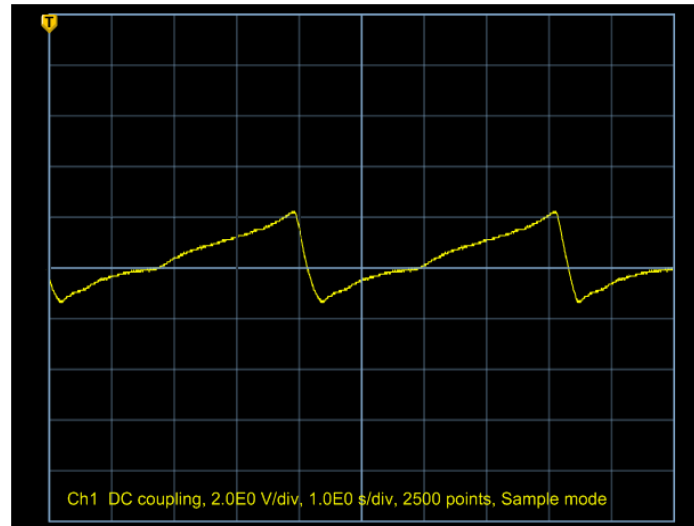
Figure 8.14 Filtered, pulsatile signal obtained for IR LED only.

From the Figure 8.14 it can be noted that band pass filter has accomplishing its tasks of removing the excess noise as well as the DC offset. The low pass stage eliminated any of the high frequency noise present in the signal including power lines 60 Hz frequency range. The high pass stage attenuated the DC offset, as well as any low frequency noise artefacts that are present due to motion of the individual user. Finally, in the amplification stage has the signal has been amplified to the range of 1-5V. The amplification was achieved by using a non-inverting operational amplifier configuration with gain of 57 times. These results corresponds well to the reference signal obtained from the same subject using commercial off-the-shelf Nonin GO2 Achieve 9570 Finger Pulse Oximeter (Nonin 2014). The obtained signal was recognized by a clinician as a correct pulse oximetry signal.

8.4.5   Cost estimation

In order to provide a rough guideline of the cost of final commercial product, an attempt has been made to summaries the total cost of a prototype system components and compare it with the lowest possible total cost of same/equivalent components, which could be used to build a similar prototype from scratch. The result of this analysis is presented in Table 8.8 below. It indicates over 85% reduction on components costs when buying components in large quantities, or by using replacement or custom build components or sub-assemblies. Base on this analysis the final commercial product cost has been roughly estimated to be no more than £100 in production.

Category	Item	Vendor	Unit Cost	Qty.	Prototype Total Cost	Final Prod. Total Cost
Platform	Wireless Sensor Platform	SHIMMER Research	£144.00	2	£288.00	£45.00[1]
	ECB Daughter Board	SHIMMER Research	£106.00	1	£106.00	£10.00[2]
	AnEx Expansion Board	SHIMMER Research	£17.00	2	£34.00	-
				Total Platform	**£428.00**	**£55.00**

Category	Item	Vendor	Unit Cost	Qty.	Prototype Total Cost	Final Prod. Total Cost
Electronics	Prototyping PCB Board	Maplin UK	£2.99	2	£2.99	£0.19
	Axial-lead Resistors	Maplin UK	£0.39	22	£8.58	£0.10
	Ceramic capacitors	Maplin UK	£0.49	5	£2.45	£0.15
	LM324N operational amplifier	Maplin UK	£1.09	2	£2.18	£0.42
	LM358N operational amplifier	Texas Instruments	£1.09	1	£1.09	£0.09
	Strip connectors	Maplin UK	£1.44	2	£2.88	£0.34
	D-sub 9 pin female connector (DE-9)	Maplin UK	£1.69	1	£1.69	£0.38
				Total Electronics	**£21.86**	**£1.67[3]**

Category	Item	Vendor	Unit Cost	Qty.	Prototype Total Cost	Final Prod. Total Cost
Sensors	Piezo Crystal Effort Sensor	S.L.P. Inc.	£193.00	1	£193.00	£1.11[4]
	Adult Reusable Pulse oximeter Flex Sensor	Nonin Medical Inc.	£99.54	1	£99.54	£15.00[5]
	Atlas Monitor ECG Lead Wires (3-Lead AHA)	Welch Allyn Inc.	£6.68	1	£6.08	£6.08
	3M ECG Wet Gel Monitoring Electrodes	3M Inc.	£0.10	3	£0.30	£0.30
	MA100 NTC Type Thermistor	GE Sensing Inc.	£9.02	1	£9.02	£3.25[6]
				Total Sensors	**£307.94**	**£25.74**
				Total System	**£757.80**	**£82.41**

Table 8.8 Cost summary for the prototype system and cost estimate for the final commercial product.

---

[1] Arduino with integrated Bluetooth Smart chip - http://redbearlab.com/blendmicro/

[2] ECG Front End - http://www.100randomtasks.com/ecg-front-end

[3] Wholesale prices for products bought in bundles of 1,000+ pcs.

[4] Prowave FS-2513P Piezo Film - http://www.prowave.com.tw/english/products/pp/film.htm

[5] Pulse Sensor, Prod. Code: G5123195FB71B1 - http://www.tinyosshop.com/

[6] Wholesale price for MA100 thermistor when bought in bundle of 1,000+ pcs.

8.5 Limitations of the research and areas of future work

By enabling the acquisition of different body parameters from commercially available third party devices as well as integration with wider telemedicine systems and external services, the presented solution aim to boost the adoption and integration of such systems in real-time everyday healthcare . In the long term, it is envisaged that wide-spread adoption of such remote monitoring systems using automated personal health assistants which can integrate multiple devices and parameters as well as autonomously make intelligent decisions, could contribute to a substantial reduction on care costs as well as on the time for delivery of care, translating consequently, in an improvement in quality of life and well-being of the population.

Currently the application of the proposed prototype framework as well as SOM data processing model to real life monitoring problems is limited due to the need for its wider and longer term testing. To fully appraise its benefits and value for everyday health monitoring, a number of different applications for many different scenarios must be developed and evaluated during all kinds of the day-to-day activities, as opposed to the ICU database evaluation, conducted during this research project. In order to address this limitation and ensure the model adoption, the future work in terms of platform and model developments will focus on:

1) Porting and extending further the platform to support any native smartphone operating system. It is envisaged that this can be achieved with a central meta code repository, which at the time of deployment, will port the application to the right platform. Technologies as well as methods to do this has been presented by Miroslav et al. (Miroslav, Fedor and Gabriel 2012).

2) Improving the error-free operation of the platform and release it to the research community for testing.

3) Development of further test beds for both the framework as well as SOM model with applications offering solutions to various medical, health and sport/well-being monitoring problems.

4) In order to fully release the potential of SOM in health monitoring the future work will focus on the implementation of the map trajectory tracking mechanism as well as the speed of ascent/descent analysis model, which will be capable to analyse the long, medium and short term trend changes and their speed.

5) Further test the integration with custom sensor devices as well as external services.

6) Extension of the capabilities and protocols for the aggregation of the real-time data from Body (BAN), Personal (PAN) and Near-me (NAN) Area Networks.

7) Extensions to the process of applications logic development through more user friendly graphical manipulation of data processing components and less through textual developments.

8) Further extensions to the inference engine model, which will introduce new signal processing components and machine learning paradigms.

8.6   Impact and applications of the proposed solution

According to recently published research and development Horizon 2020 Work Programme 2014-2015 for the European Union (European Commision 2014) the health and well-being in an ageing society is one of the Europe's key socio-economic challenges. It focuses on ICT for disease prediction, early diagnosis, prevention, minimally invasive treatment, overall disease management and support to healthy lifestyles. As part of the target outcomes for this challenge the Intelligent Personal Health Systems (PHS) for remote management of diseases, treatment, rehabilitation or analysis of multi-parametric data, amongst others, are expected. However such systems could also apply to different domains, where a person's physical condition is an important factor, such as sport, computer games, military, automotive etc. Current systems are usually single problem tailored solutions that sense, transmit and process single or few parameters and their deployment requires each time a considerable amount of effort, cost and resources. Recent advances and popularisation of smartphones, miniaturisation and new sensor devices have put such systems at the cross-road, demanding for more unified, more integrative, more intelligent and cognitive systems that can get easy and quickly assembled and deployed to solve particular research problems or answer commercial needs. Successful implementation of such systems requires rethinking and redesign/adoption of current standards in networking and service infrastructures as well as in terms of cognitive systems what also address other challenges of an EC Work Programme. The aim of this project was to come across this needs and build a new proof of concept tool that could challenge current thinking of biofeedback information and its usage as well as opening up new, innovative research paths in many disciplines.

The framework presented in this work is expected to addresses elements that are common for current mobile health monitoring systems such as wireless sensing, data filtering and processing, as well as interconnection of external services, amongst others, but taken into a new integrative level. This will eliminate redundant tasks and therefore reduce cost, time and effort when developing such applications in the future as well as minimizing their "time-to-market". This, on the other hand, will also mean that the applications will be ready for deployment in a shorter period increasing the benefits for the end users. Having a framework such as the one proposed here will allow researchers and developers to focus more on the knowledge intrinsic to the data rather than on technical programming details. With this framework proposal it is envisage to offer researchers and developers a highly methodological and technical tool for rapid applications development with bio-signals monitoring. The main impact of this framework comes from its application to new systems on several domains all of which contribute directly or indirectly to the well-being of the end users. Among these, the medical field is the one offering the highest impact. Given the possibilities offered by this framework, applications targeting the elderly as well as general improved health services, are envisaged to have direct impact on the population of well-developed countries which are

recognised to face future problems. This is due to, on one hand an ageing population with increased life expectancy and demographical valley; while on the other hand, are being presented with a strain on health services as a result of high-rising costs of healthcare liabilities and the current economic situation.

The presented model enables adaptive monitoring of patients using patient specific models. This provides a more effective approach to identifying potential health risks and specific clinical symptoms of an individual, particularly when compared to the conventional, population-based diagnostic approaches currently used. It has been shown that acquisition and analysis of multiple vital sign parameters from a single patient in real time, together with continuous adaptation of the level of detail of their analysis enables a more precise understanding of the patient's health status and eventual diagnosis goals. By designing this highly adaptable, distributed model, capable of self-adjusting over time based on historical vital sign measurements, individuals are able to keep physically active, detection and early notification of potential illness risks is improved. Also more accurate treatments "on-the-go" will be possible whilst admissions to hospitals would be reduced. The proposed solution is expected to continue to support illness prevention and early detection, enabling management of wellness rather than illness.

Overall, the ubiquitous, user-tailored and interconnected systems built using the proposed framework will challenge obstacles faced by current problem dedicated systems, and provide a unified, non-intrusive, and low cost solutions for monitoring of full range of biomedical parameters. This will allow focusing on prevention, early detection and management of wellness rather than illness. Another important impact envisaged could be its application on the serious games and commercial games fields. The tremendous impact and growing relevance that the game industry has played in recent years in providing valued entertainment as well as invaluable potentials for training and education will only be strengthened by the addition of the user's real-time biofeedback information. Thus wider adoption of the proposed framework for applications using real-time biofeedback information in medicine, sport and games will further improve the quality of services to the end users by enhancing their quality of life, and well-being.

This research has focused on mobile application of Artificial Intelligence algorithms to vital signs monitoring, through multi-parametric, user-adaptable model for ubiquitous well-being monitoring, and has successfully achieved its objectives. Given the potential benefits of the solution provided, the author will expand on the presented data processing model test beds as well as will continue with the middleware design and implementation, what should eventually lead to realise of the framework to the public.

References

A&D Medical Inc (n.d.) *Wireless Automatic Blood Pressure Monitor* [online] available from <http://www.andmedical.com/and_med.nsf/index> [September 2009]

Abibullaev, B. and Seo, H. D. (2011) 'A New Qrs Detection Method Using Wavelets and Artificial Neural Networks.' *Journal of Medical Systems* 35, (4) 683-691 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-77954798584&partnerID=40&md5=689999de89d207980d4d6d0795045d6b>

Adnane, M., Jiang, Z. and Choi, S. (2009) 'Development of Qrs Detection Algorithm Designed for Wearable Cardiorespiratory System.' *Computer Methods and Programs in Biomedicine* 93, (1) 20-31 available from <http://www.sciencedirect.com/science/article/pii/S0169260708001879>

Ahlsén, M., Asanin, S., Kool, P., Rosengren, P. and Thestrup, J. (2012) 'Service-Oriented Middleware Architecture for Mobile Personal Health Monitoring.' In *Wireless Mobile Communication and Healthcare.* vol. 83 ed. by Nikita, K., Lin, J., Fotiadis, D. and Arredondo Waldmeyer, M.-T.: Springer Berlin Heidelberg: 305-312. Available from <http://dx.doi.org/10.1007/978-3-642-29734-2_42>

Akinduko, A. A. and Mirkes, E. M. (2012) 'Initialization of Self-Organizing Maps: Principal Components Versus Random Initialization. A Case Study.' *arXiv preprint arXiv:1210.5873*

Alemdar, H. and Ersoy, C. (2010) 'Wireless Sensor Networks for Healthcare: A Survey.' *Computer Networks* 54, (15) 2688-2710 available from <http://www.sciencedirect.com/science/article/pii/S1389128610001398>

Alemzadeh, H., Zhanpeng, J., Kalbarczyk, Z. and Iyer, R. K. (2011) 'An Embedded Reconfigurable Architecture for Patient-Specific Multi-Paramater Medical Monitoring.' *Engineering in Medicine and Biology Society,EMBC, 2011 Annual International Conference of the IEEE.* Aug. 30 2011-Sept. 3 2011

Alive Technologies Ltd. (n.d.) [online] available from <http://www.alivetec.com/> [August 2009]

Ambu Sleepmate Inc. (n.d.) [online] available from <http://www.sleepmate.com/index.jsp> [June 2009]

American Educational Research Association, American Psychological Association, National Council on Measurement in Education and Joint Committee on Standards for Educationa (1999) *Standards for Educational and Psychological Testing.* Washington, DC: American Educational Research Association

American National Standards Institute (1998) *Ansi/Aami Ec57: Testing and Reporting Performance Results of Cardiac Rhythm and St Segment Measurement Algorithms.* Available from <http://www.aami.org>

Andersen, M. (2006) *Human Body Temperature at a Variety of Room Temperatures.*: MIT OpenCourseWare

Andre, D., Pelletier, R., Farringdon, J., Safier, S., Talbott, W., Stone, R., Vyas, N., Trimble, J., Wolf, D., Vishnubhatla, S., Boehmke, S., Stivoric, J. and Teller, A. (2006) 'The Development of the Sensewear Armband, a Revolutionary Energy Assessment Device to Assess Physical Activity and Lifestyle.' [online]. Available from <http://www.bodymedia.com/Professionals/Whitepapers/The-Development-of-the-SenseWear-armband->

Andreao, R. V., Dorizzi, B. and Boudy, J. (2006) 'Ecg Signal Analysis through Hidden Markov Models.' *Biomedical Engineering, IEEE Transactions on* 53, (8) 1541-1549

Angner, E., Ray, M. N., Saag, K. G. and Allison, J. J. (2009) 'Health and Happiness among Older Adults: A Community-Based Study.' *Journal of Health Psychology* 14, (4) 503-512

Anguita, D., Boni, A., Ridella, S., Rivieccio, F. and Sterpi, D. (2005) 'Theoretical and Practical Model Selection Methods for Support Vector Classifiers.' In *Support Vector Machines: Theory and Applications.* vol. 177 ed. by Wang, L.: Springer Berlin Heidelberg: 159-179. Available from <http://dx.doi.org/10.1007/10984697_7>

Anguita, D., Ghio, A., Greco, N., Oneto, L. and Ridella, S. (2010) 'Model Selection for Support Vector Machines: Advantages and Disadvantages of the Machine Learning Theory.' *Neural Networks (IJCNN), The 2010 International Joint Conference on.* 18-23 July 2010

Anliker, U., Ward, J. A., Lukowicz, P., Troster, G., Dolveck, F., Baer, M., Keita, F., Schenker, E. B., Catarsi, F., Coluccini, L., Belardinelli, A., Shklarski, D., Alon, M., Hirt, E., Schmid, R. and Vuskovic, M. (2004) 'Amon: A Wearable Multiparameter Medical Monitoring and Alert System.' *Information Technology in Biomedicine, IEEE Transactions on* 8, (4) 415-427

Aquilani, F., Balsamo, S. and Inverardi, P. (2001) 'Performance Analysis at the Software Architectural Design Level.' *Performance Evaluation* 45, (2–3) 147-178 available from <http://www.sciencedirect.com/science/article/pii/S0166531601000359>

Association for the Advancement of Medical Instrumentation (2012) *Testing and Reporting Performance Results of Cardiac Rhythm and St Segment Measurement Algorithms.* AAMI/ANSI Standard EC57

Atallah, L., Lo, B. and Yang, G.-Z. (2012) 'Can Pervasive Sensing Address Current Challenges in Global Healthcare?' *Journal of Epidemiology and Global Health* 2, (1) 1-13 available from <http://www.sciencedirect.com/science/article/pii/S2210600612000020>

Augustyniak, P. (2005) 'Adaptive Wearable Vital Signs Monitor for Home Care and Sports ' In Kurzyński, M., Puchała, E., Wozniak, M. and Żołnierek, A. (ed.) *Computer Recognition Systems vol. 30/2005*, *4-th CORES'2005* Springer Verlag

Avritzer, A. and Weyuker, E. (1999) 'Metrics to Assess the Likelihood of Project Success Based on Architecture Reviews.' *Empirical Software Engineering* 4, (3) 199-215 available from <http://dx.doi.org/10.1023/A%3A1009826509846>

Babar, M. A. and Gorton, I. (2004) 'Comparison of Scenario-Based Software Architecture Evaluation Methods.' *Software Engineering Conference, 2004. 11th Asia-Pacific.* 30 Nov.-3 Dec. 2004

Baker, F. C. and Driver, H. S. (2007) 'Circadian Rhythms, Sleep, and the Menstrual Cycle.' *Sleep Medicine* 8, (6) 613-622 available from <http://www.sciencedirect.com/science/article/B6W6N-4NBRG0R-1/2/e54cdedcc79cb90acfb62b9d61b3f6c2>

Barill, T. (2005) 'An Ecg Primer.' In *Six Second Ecg : A Practical Guide to Basic and 12 Lead Ecg Interpretation. .* Vancouver BC: Nursecom Educational Technologies: 63-100. Available from <http://www.nursecom.com/ECGprimer.pdf>

Barlow, J., Singh, D., Bayer, S. and Curry, R. (2007) 'A Systematic Review of the Benefits of Home Telecare for Frail Elderly People and Those with Long-Term Conditions.' *Journal of Telemedicine and Telecare* 13, (4) 172-179 available from <http://jtt.sagepub.com/content/13/4/172.abstract>

Bengtsson, P., Lassing, N., Bosch, J. and Van Vliet, H. (2004) 'Architecture-Level Modifiability Analysis (Alma).' *Journal of Systems and Software* 69, (1–2) 129-147 available from <http://www.sciencedirect.com/science/article/pii/S0164121203000803>

Benitez, D., Gaydecki, P. A., Zaidi, A. and Fitzpatrick, A. P. (2001) 'The Use of the Hilbert Transform in Ecg Signal Analysis.' *Computers in Biology and Medicine* 31, (5) 399-406 available from <http://www.sciencedirect.com/science/article/B6T5N-43VRMC8-7/2/31dfc9ed43af5a35bedc2f92b518a0b9>

Bergmo, T. S. (2010) 'Economic Evaluation in Telemedicine – Still Room for Improvement.' *Journal of Telemedicine and Telecare* 16, (5) 229-231 available from <http://jtt.rsmjournals.com/content/16/5/229.abstract>

Bickley, L. S. and Szilagyi, P. G. (2007) *Bates' Guide to Physical Examination and History Taking.* 9th edn.: Lippincott Williams & Wilkins

Blobel, B. G. (2007) 'Educational Challenge of Health Information Systems' Interoperability.' *Methods of Information in Medicine* 46, (1) 52-56 available from <http://europepmc.org/abstract/MED/17224981>

Bluecove Team (2008) *Bluecove Jsr-82 Implementation* [online] available from <http://bluecove.org/> [10 2010]

Bluetooth (n.d.) [online] available from <http://www.bluetooth.com/bluetooth/> [February 2009]

Bluetooth Sig (2007) *Bluetooth Specification Version 2.1 + Edr* [online] vol. 0

Body Media Inc. (2009) [online] available from <http://www.bodymedia.com/> [August 2009]

Bottou, L. and Lin, C.-J. (2007) '**Support Vector Machine Solvers**.' In *Large Scale Kernel Machines.* ed. by Bottou, L., Chapelle, O., D., D. and J., W.: MIT Press

Bradner, S. (1997) *Key Words for Use in Rfcs to Indicate Requirement Levels* [online] available from <http://xml.resource.org/public/rfc/html/rfc2119.html> [10 2010]

Braun, S. (1990) 'Respiratory Rate and Pattern.' In *Clinical Methods: The History, Physical, and Laboratory Examinations.* 3 edn. ed. by Walker HK, Hall WD and Hurst JW Boston: Butterworth-Heinemann

Brown, M., Grundy, W., Lin, D., Christianini, N., Sugnet, C., Ares, M. J. and Haussler, D. (1999) *Support Vector Machine Classification of Microarray Gene Expression Data.* Santa Cruz, CA: Department of Computer Science, University of California

Burges, C. J. C. (1998) 'A Tutorial on Support Vector Machines for Pattern Recognition.' *Data Mining and Know ledge Discovery* 2, (2) 121-167

Burns, A., Greene, B. R., Mcgrath, M. J., O'shea, T. J., Kuris, B., Ayer, S. M., Stroiescu, F. and Cionca, V. (2010) 'Shimmer - a Wireless Sensor Platform for Noninvasive Biomedical Research.' *Sensors Journal, IEEE* 10, (9) 1527-1534

Busuoli, M., Gallelli, T., Haluzík, M., Fabián, V., Novák, D. and Štěpánková, O. (2007) 'Entertainment and Ambient: A New Oldes' View.' In *Universal Access in Human-Computer Interaction. Applications and Services.* vol. 4556 ed. by Stephanidis, C.: Springer Berlin Heidelberg: 511-519. Available from <http://dx.doi.org/10.1007/978-3-540-73283-9_57>

Campana, F., Moreno, A., Riaño, D. and Varga, L. (2008) 'K4care: Knowledge-Based Homecare E-Services for an Ageing Europe.' In *Agent Technology and E-Health.* ed. by Annicchiarico, R., Cortés, U. and Urdiales, C.: Birkhäuser Basel: 95-115. Available from <http://dx.doi.org/10.1007/978-3-7643-8547-7_6>

Carroll, R., Cnossen, R., Schnell, M. and Simons, D. (2007) 'Continua: An Interoperable Personal Healthcare Ecosystem.' *Pervasive Computing, IEEE* 6, (4) 90-94

Cecchet, E., Marguerite, J. and Zwaenepoel, W. (2002) 'Performance and Scalability of Ejb Applications.' *SIGPLAN Not.* 37, (11) 246-261

Chan, M., Estève, D., Fourniols, J.-Y., Escriba, C. and Campo, E. (2012) 'Smart Wearable Systems: Current Status and Future Challenges.' *Artificial Intelligence in Medicine* 56, (3) 137-156 available from <http://www.sciencedirect.com/science/article/pii/S0933365712001182>

Changzhi, L., Cummings, J., Lam, J., Graves, E. and Wenhsing, W. (2009) 'Radar Remote Monitoring of Vital Signs.' *Microwave Magazine, IEEE* 10, (1) 47-56

Chen, M. M., Bush, J. W. and Patrick, D. L. (1975) 'Social Indicators for Health Planning and Policy Analysis.' *Policy Sciences* 6, (1) 71-89 available from <http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=16625643&site=ehost-live>

Chen, S.-W., Chen, H.-C. and Chan, H.-L. (2006) 'A Real-Time Qrs Detection Method Based on Moving-Averaging Incorporating with Wavelet Denoising.' *Computer Methods and Programs in Biomedicine* 82, (3) 187-195 available from <http://www.sciencedirect.com/science/article/pii/S0169260705002592>

Cherkassky, V., Xuhui, S., Mulier, F. M. and Vapnik, V. N. (1999) 'Model Complexity Control for Regression Using Vc Generalization Bounds.' *Neural Networks, IEEE Transactions on* 10, (5) 1075-1089

Chisholm, H. (1911) *Animal Heat: Diurnal Variation in Body Temperature.* [48-50] New York :: Encyclopaedia Britannica

Chobanian, A. V., Bakris, G. L., Black, H. R., Cushman, W. C., Green, L. A., Izzo, J. L., Jr., Jones, D. W., Materson, B. J., Oparil, S., Wright, J. T., Jr., Roccella, E. J. and The National High Blood Pressure Education Program Coordinating, C. (2003) 'Seventh Report of the Joint National Committee on Prevention, Detection, Evaluation, and Treatment of High Blood Pressure.' *Hypertension* 42, (6) 1206-1252 available from <http://hyper.ahajournals.org/cgi/content/abstract/42/6/1206>

Christov, Ii and Daskalov, I. K. (1999) 'Filtering of Electromyogram Artifacts from the Electrocardiogram.' *Med Eng Phys* 21, (10) 731-736

Christov, I., Gómez-Herrero, G., Krasteva, V., Jekova, I., Gotchev, A. and Egiazarian, K. (2006) 'Comparative Study of Morphological and Time-Frequency Ecg Descriptors for Heartbeat Classification.' *Medical Engineering & Physics* 28, (9) 876-887 available from <http://www.sciencedirect.com/science/article/pii/S135045330600004X>

Clifford, G., Shoeb, A., Mcsharry, P. and Janz, B. (2005) 'Model-Based filtering, Compression and Classification of the Ecg.' *Intern. J. Bioelectromagnetism* 7, (1) 158–161

Cohen, K. P., Tompkins, W. J., Djohan, A., Webster, J. G. and Hu, Y. H. (1995) 'Qrs Detection Using a Fuzzy Neural Network.' *vol. 17.* Available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0029428853&partnerID=40&md5=2246503b7fbec5b215f80c40fc94e0a4>

Cortes, C. and Vapnik, V. (1995) 'Support-Vector Networks.' *Machine Learning* 20, (3) 273-297 available from <http://dx.doi.org/10.1007/BF00994018>

Coyle, S., Lau, K. T., Moyna, N., O''Gorman, D., Diamond, D., Di Francesco, F., Costanzo, D., Salvo, P., Trivella, M. G., De Rossi, D. E., Taccini, N., Paradiso, R., Porchet, J. A., Ridolfi, A., Luprano, J., Chuzel, C., Lanier, T., Revol-Cavalier, F., Schoumacker, S., Mourier, V., Chartier, I., Convert, R., De-Moncuit, H. and Bini, C. (2010) 'Biotex - Biosensing Textiles for Personalised Healthcare Management.' *Information Technology in Biomedicine, IEEE Transactions on* 14, (2) 364-370

Coyle, S., Morris, D., Lau, K.-T., Diamond, D., Taccini, N., Constanzo, D., Salvo, P., Di Francesco, F., Trivella, M. G., Porchet, J. A. and Luprano, J. (2009) 'Textile Sensors to Measure Sweat Ph and Sweat-Rate During Exercise.' *Pervasive Health 2009.* 1-3 April Held at London, UK: IEEE

Craddock, I. (2015) *Sphere - a Sensor Platform for Healthcare in a Residential Environment* [online] available from <http://www.irc-sphere.ac.uk/>

Cranston, W. I., Gerbrandy, J. and Snell, E. S. (1954) 'Oral, Rectal and Oesophageal Temperatures and Some Factors Affecting Them in Man.' *The Journal of Physiology* 126, (2) 347-358

Cretikos, M., Bellomo, R., Hillman, K., Chen, J., Finfer, S. and Flabouris, A. 'Respiratory Rate: The Neglected Vital Sign.' (0025-729X (Print))

Crisp, R. (2001) 'Well-Being.' *Stanford Encyclopedia of Philosophy* [online]. Available from <http://plato.stanford.edu/entries/well-being/>

Cristianini, N. and Shawe-Taylor, J. (2000) *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods.* Cambridge University Press

Crocker, D. (ed.) (1997) *Rfc 2234: Augmented Bnf for Syntax Specifications: Abnf.* Available from <http://www.apps.ietf.org/rfc/rfc2234.html>

Crossbow Technology Inc. (n.d.) [online] available from <http://www.xbow.com/Products/productdetails.aspx?sid=156> [September 2009]

Curone, D., Secco, E. L., Tognetti, A., Loriga, G., Dudnik, G., Risatti, M., Whyte, R., Bonfiglio, A. and Magenes, G. (2010) 'Smart Garments for Emergency Operators: The Proetex Project.' *Information Technology in Biomedicine, IEEE Transactions on* 14, (3) 694-701

Daamouche, A., Hamami, L., Alajlan, N. and Melgani, F. (2011) 'A Wavelet Optimization Approach for Ecg Signal Classification.' *Biomedical Signal Processing and Control* available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-79960653847&partnerID=40&md5=526c4bdd16a79f67b6f0a68a53b269b7>

Dabiri, F., Noshadi, H., Hagopian, H., Massey, T. and Sarrafzadeh, M. (2007) *Lightweight Medical Bodynets*. Proceedings of the ICST 2nd international conference on Body area networks. Florence, Italy: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)

Danna, K. and Griffin, R. W. (1999) 'Health and Well-Being in the Workplace: A Review and Synthesis of the Literature.' *Journal of Management* 25, (3) 357-384 available from <http://jom.sagepub.com/cgi/content/abstract/25/3/357>

Daubechies, I. (1990) 'The Wavelet Transform, Time-Frequency Localization and Signal Analysis.' *Information Theory, IEEE Transactions on* 36, (5) 961-1005

De Chazal, P., O'dwyer, M. and Reilly, R. B. (2004) 'Automatic Classification of Heartbeats Using Ecg Morphology and Heartbeat Interval Features.' *IEEE Transactions on Biomedical Engineering* 51, (7) 1196-1206 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-2942709662&partnerID=40&md5=b41630fee6b38012a5009dc0fac90945>

De Chazal, P. and Reilly, R. B. (2006) 'A Patient-Adapting Heartbeat Classifier Using Ecg Morphology and Heartbeat Interval Features.' *IEEE Transactions on Biomedical Engineering* 53, (12) 2535-2543 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-33845864226&partnerID=40&md5=22d592f44d405c9a36c510ae31d3e8f7>

Deshpande, A. K. S. and Mckibbon, J. A. (2008) *Real-Time (Synchronous) Telehealth in Primary Care:Systematic Review of Systematic Reviews.* Ottawa: Canadian Agency for Drugs and Technologies in Health (CADTH)

Devaul, R. W., Sung, M., Gips, J. and Pentland, S. (2003) *Mithril 2003: Applications and Architecture*. ISWC 2003,. USA

Di Rienzo, M., Rizzo, F., Parati, G., Brambilla, G., Ferratini, M. and Castiglioni, P. (2005) 'Magic System: A New Textile-Based Wearable Device for Biological Signal Monitoring. Applicability in Daily Life and Clinical Setting.' *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. 2005

Dickens, B. M. and Cook, R. J. (2006) 'Legal and Ethical Issues in Telemedicine and Robotics.' *International journal of gynaecology and obstetrics: the official organ of the International Federation of Gynaecology and Obstetrics* 94, (1) 73-78 available from <http://linkinghub.elsevier.com/retrieve/pii/S002072920600186X?showall=true>

Diener, E. (2000) 'Subjective Well-Being: The Science of Happiness, and a Proposal for a National Index.' *American Psychologist* 55, (1) 34-43

Diener, E. and Seligman, M. E. P. (2004) 'Beyond Money.' *Psychological Science in the Public Interest* 5, (1) 1-31 available from <10.1111/j.0963-7214.2004.00501001.x http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=13436512&site=ehost-live>

Dobrica, L. and Niemela, E. (2002) 'A Survey on Software Architecture Analysis Methods.' *Software Engineering, IEEE Transactions on* 28, (7) 638-653

Doğan, B. and Korürek, M. (2012) 'A New Ecg Beat Clustering Method Based on Kernelized Fuzzy C-Means and Hybrid Ant Colony Optimization for Continuous Domains.' *Applied Soft Computing* 12, (11) 3442-3451 available from <http://www.sciencedirect.com/science/article/pii/S1568494612003079>

Dokur, Z. and Ölmez, T. (2001) 'Ecg Beat Classification by a Novel Hybrid Neural Network.' *Computer Methods and Programs in Biomedicine* 66, (2-3) 167-181 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0034909765&partnerID=40&md5=db8b38f13d904b5c674104b6470cfc07>

Dong, T., Kumar, D., Duvur, S. and Torbjornsen, O. (2004) 'Availability Measurement and Modeling for an Application Server.' *Dependable Systems and Networks, 2004 International Conference on*. 28 June-1 July 2004

Dotsinsky, I. A. and Mihov, G. S. (2008) 'Tremor Suppression in Ecg.' *Biomed Eng Online* 7, 29

Droitcour, A. (2006) *Non-Contact Measurement of Heart and Respiration Rates with a Single-Chip Microwave Doppler Radar* Unpublished thesis, Stanford University

Dutta, S., Chatterjee, A. and Munshi, S. (2010) 'Correlation Technique and Least Square Support Vector Machine Combine for Frequency Domain Based Ecg Beat Classification.' *Medical Engineering & Physics* 32, (10) 1161-1169 available from <http://www.sciencedirect.com/science/article/pii/S1350453310001761>

Eisenhauer, M., Rosengren, P. and Antolin, P. (2009) 'A Development Platform for Integrating Wireless Devices and Sensors into Ambient Intelligence Systems.' *Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2009. SECON Workshops '09. 6th Annual IEEE Communications Society Conference on*. 22-26 June 2009

Ekeland, A. G., Bowes, A. and Flottorp, S. (2010) 'Effectiveness of Telemedicine: A Systematic Review of Reviews.' *International Journal of Medical Informatics* 79, (11) 736-771 available from <http://www.sciencedirect.com/science/article/pii/S1386505610001504>

Engin, M. (2004) 'Ecg Beat Classification Using Neuro-Fuzzy Network.' *Pattern Recognition Letters* 25, (15) 1715-1722 available from <http://www.sciencedirect.com/science/article/pii/S0167865504001588>

Enough Software (2004) *J2me Polish, the Quasi Standard for Mobile Java and Cross-Platform Development* [online] available from <http://www.enough.de/products/j2me-polish/> [October 2011]

European Commision (2014) *Horizon 2020 Work Programme 2014 – 2015: 8. Health, Demographic Change and Wellbeing.*

Horizon 2020 - Work Programme 2014 – 2015 (2014). Available from <http://ec.europa.eu/research/participants/data/ref/h2020/wp/2014_2015/main/h2020-wp1415-intro_en.pdf>

Evanczuk, S. (2013) 'Ant/Ant+ Solutions Speed Low-Power Wireless Design.' *Wireless solutions* [online]. Available from <http://www.digikey.com/us/en/techzone/wireless/resources/articles/ant-ant-solutions-speed-low-power-wireless-design.html> [April 2013]

Ferraris, F., Grimaldi, U. and Parvis, M. (1995) 'Procedure for Effortless in-Field Calibration of Three-Axis Rate Gyros and Accelerometers.' *Sens. Mater.* 7, 311-330

Floudas, C. and Visweswaran, V. (1995) 'Quadratic Optimization.' In *Handbook of Global Optimization.* vol. 2 ed. by Horst, R. and Pardalos, P.: Springer US: 217-269. Available from <http://dx.doi.org/10.1007/978-1-4615-2025-2_5>

Fluke Biomedical (2010) *Fluke Mps450 Patient Simulator* [online] available from <http://www.flukebiomedical.com/Biomedical/usen/Biomedical-Test/Simulators-&-Controllers/MPS450.htm?PID=56424> [December 2010]

Freescale Semiconductor Inc. (2008) *Mma7361l ±1.5g, ±6g Three Axis Low-G Micromachined Accelerometer* [online] available from <http://www.freescale.com/files/sensors/doc/data_sheet/MMA7361L.pdf> [December

Friedsam, H. J. and Martin, H. W. (1963) 'A Comparison of Self and Physicians' Health Ratings in an Older Population ' *Journal of Health and Human Behavior* 4, (3) 179-183

Fugal, D. L. (2009) *Conceptual Wavelets in Digital Signal Processing: An in-Depth Practical Approach for the Non-Mathematician.* San Diego, California: Space & Signals Technical Publishing

Fung, P., Dumont, G., Ries, C., Mott, C. and Ansermino, M. (2004) 'Continuous Noninvasive Blood Pressure Measurement by Pulse Transit Time.' *vol. 1, Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International Conference of the IEEE.* 1-5 Sept. 2004

Gacek, A. (2011) 'Preprocessing and Analysis of Ecg Signals - a Self-Organizing Maps Approach.' *Expert Systems with Applications* 38, (7) 9008-9013 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-79952443365&partnerID=40&md5=72a7b38de2dd7d018d838795606e1d9f>

Gao, H., Mcdonnell, A., Harrison, D. A., Moore, T., Adam, S., Daly, K., Esmonde, L., Goldhill, D. R., Parry, G. J., Rashidian, A., Subbe, C. P. and Harvey, S. (2007) 'Systematic Review and Evaluation of Physiological Track and Trigger Warning Systems for Identifying at-Risk Patients on the Ward.' *Intensive Care Med* 33, (4) 667-679

Gay, D., Levis, P. and Culler, D. (2007) 'Software Design Patterns for Tinyos.' *ACM Trans. Embed. Comput. Syst.* 6, (4) 22

Ge Healthcare (2010) *Mac 3500 Ecg Analysis System* [online] available from <http://www.gehealthcare.com/euen/cardiology/products/diagnostic_ecg/resting_stress/mac_3500/index.html> [December

Ge Industrial Sensing (2006) 'Ntc Type Ma Thermometrics Biomedical Chip Thermistors.' [online]. Available from <http://www.ge-mcs.com/en/temperature/medical-assemblies/ntc-type-ma.html> [20/07/2013]

Gentag Inc. (n.d.) [online] available from <http://www.gentag.com/> [June 2009]

Genton, M. G. (2002) 'Classes of Kernels for Machine Learning: A Statistics Perspective.' *J. Mach. Learn. Res.* 2, 299-312

Goldberger, A. L., Amaral, L. a. N., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K. and Stanley, H. E. (2000) 'Physiobank, Physiotoolkit, and Physionet : Components of a New Research Resource for Complex Physiologic Signals.' *Circulation* 101, (23) e215-220 available from <http://circ.ahajournals.org/cgi/content/abstract/101/23/e215>

Goletsis, Y., Papaloukas, C., Fotiadis, D. I., Likas, A. and Michalis, L. K. (2004) 'Automated Ischemic Beat Classification Using Genetic Algorithms and Multicriteria Decision Analysis.' *Biomedical Engineering, IEEE Transactions on* 51, (10) 1717-1725

Gomes, P. R., Soares, F. O., Correia, J. H. and Lima, C. S. (2010) 'Ecg Data-Acquisition and Classification System by Using Wavelet-Domain Hidden Markov Models.' *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE.* Aug. 31 2010-Sept. 4 2010

Gomez, C., Oller, J. and Paradells, J. (2012) 'Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology.' *Sensors* 12, (9) 11734-11753 available from <http://www.mdpi.com/1424-8220/12/9/11734>

Gorgas, D. L. (2004) 'Vital Signs and Patient Monitoring Techniques.' In *Clinical Procedures in Emergency Medicine.* 4 edn. ed. by Roberts, J. R. and Hedges, J. Philadelphia: Saunders: 3-28

Grossman, P. (2004) 'The Lifeshirt: A Multi-Function Ambulatory System Monitoring Health, Disease, and Medical Intervention in the Real World.' *Studies in health technology and informatics* 108, 133-141 available from <http://view.ncbi.nlm.nih.gov/pubmed/15718639>

Güler, İ. and Übeylı̇, E. D. (2005) 'Ecg Beat Classifier Designed by Combined Neural Network Model.' *Pattern Recognition* 38, (2) 199-208 available from <http://www.sciencedirect.com/science/article/pii/S0031320304002766>

Gungor, V. C. and Hancke, G. P. (2009) 'Industrial Wireless Sensor Networks: Challenges, Design Principles, and Technical Approaches.' *Industrial Electronics, IEEE Transactions on* 56, (10) 4258-4265

Guo, D. G., Tay, F. E. H., Xu, L., Yu, L. M., Nyan, M. N., Chong, F. W., Yap, K. L. and Xu, B. (2009) 'Characterization and Fabrication of Novel Micromachined Electrode for Bsn-Based Vital Signs Monitoring System.' *Wearable and Implantable Body Sensor Networks, 2009. BSN 2009. Sixth International Workshop on*. 3-5 June 2009

Guyon, I. and Elisseeff, A. (2003) 'An Introduction to Variable and Feature Selection.' *J. Mach. Learn. Res.* 3, 1157-1182

Guyton, A. and Hall, J. (2006) *Textbook of Medical Physiology*. 11 edn. Philadelphia: Elsevier Saunders

Haahr, R. G., Duun, S., Thomsen, E. V., Hoppe, K. and Branebjerg, J. (2008) 'A Wearable Electronic Patch for Wireless Continuous Monitoring of Chronically Diseased Patients.' *Medical Devices and Biosensors, 2008. ISSS-MDBS 2008. 5th International Summer School and Symposium on*. 1-3 June 2008

Haataja, K. and Toivanen, P. (2010) 'Two Practical Man-in-the-Middle Attacks on Bluetooth Secure Simple Pairing and Countermeasures.' *Wireless Communications, IEEE Transactions on* 9, (1) 384-392

Habetha, J. (2006) 'The Myheart Project - Fighting Cardiovascular Diseases by Prevention and Early Diagnosis.' *vol. Supplement, Engineering in Medicine and Biology Society, 2006. EMBS '06. 28th Annual International Conference of the IEEE*. Aug. 30 2006-Sept. 3 2006

Hadj Slimane, Z.-E. and Naït-Ali, A. (2010) 'Qrs Complex Detection Using Empirical Mode Decomposition.' *Digital Signal Processing* 20, (4) 1221-1228 available from <http://www.sciencedirect.com/science/article/pii/S105120040900195X>

Halliday, A. J., Moulton, S. E., Wallace, G. G. and Cook, M. J. (2012) 'Novel Methods of Antiepileptic Drug Delivery — Polymer-Based Implants.' *Advanced Drug Delivery Reviews* 64, (10) 953-964 available from <http://www.sciencedirect.com/science/article/pii/S0169409X1200124X>

Hamilton, P. S. and Tompkins, W. J. (1986) 'Quantitative Investigation of Qrs Detection Rules Using the Mit/Bih Arrhythmia Database.' *IEEE Trans Biomed Eng* 33, (12) 1157-1165 available from <http://view.ncbi.nlm.nih.gov/pubmed/3817849>

Han, C.-C., Kumar, R., Shea, R., Kohler, E. and Srivastava, M. (2005) *A Dynamic Operating System for Sensor Nodes.* Proceedings of the 3rd international conference on Mobile systems, applications, and services. Seattle, Washington: ACM

Hass, O. C. L. and Burnham, K. J. (ed.) (2008) *Intelligent and Adaptive Systems in Medicine.* Taylor & Francis

Hauer, J.-H., Levis, P., Handziski, V. and Gay, D. (2006) *Tep 101: Analog-to-Digital Converters (Adcs)* [online] available from <http://www.tinyos.net/tinyos-2.1.0/doc/html/tep101.html>

Healthfrontier Inc (n.d.) *Ecganywhere* [online] available from <http://www.healthfrontier.com/> [September 2009]

Heart of England Nhs Foundation Trust (2011) *Adult Modified Early Warning Score (Mews) Policy & Escalation Pathway V3.0.* Available from <http://www.heartofengland.nhs.uk/wp-content/uploads/MEWS.pdf>

Hensel, B. K., Demiris, G. and Courtney, K. L. (2006) 'Defining Obtrusiveness in Home Telehealth Technologies: A Conceptual Framework.' *Journal of the American Medical Informatics Association* 13, (4) 428-431 available from <http://jamia.bmj.com/content/13/4/428.abstract>

Heskes, T. and Kappen, B. (1991) 'Neural Networks Learning in a Changing Environment.' *vol. i*, *Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on.* 8-14 Jul 1991

Hexoskin (2012) *Hexoskin - Wearable Body Metrics* [online] available from <http://www.hexoskin.com/en> [03/07/2013

Higgins, Y., Maries-Tillott, C., Quinton, S. and Richmond, J. (2008) 'Promoting Patient Safety Using an Early Warning Scoring System.' *Nursing Standard* 22, (44) 35-40 available from <http://dx.doi.org/10.7748/ns2008.07.22.44.35.c6586> [2014/06/17]

Hoekema, R., Uijen, G. J. H. and Van Oosterom, A. (2001) 'Geometrical Aspects of the Interindividual Variability of Multilead Ecg Recordings.' *Biomedical Engineering, IEEE Transactions on* 48, (5) 551-559

Homaeinezhad, M. R., Atyabi, S. A., Tavakkoli, E., Toosi, H. N., Ghaffari, A. and Ebrahimpour, R. (2012) 'Ecg Arrhythmia Recognition Via a Neuro-Svm–Knn Hybrid Classifier with Virtual Qrs Image-Based Geometrical Features.' *Expert Systems with Applications* 39, (2) 2047-2058 available from <http://www.sciencedirect.com/science/article/pii/S0957417411011365>

Hood, L. and Friend, S. H. (2011) 'Predictive, Personalized, Preventive, Participatory (P4) Cancer Medicine.' *Nat Rev Clin Oncol* 8, (3) 184-187 available from <http://dx.doi.org/10.1038/nrclinonc.2010.227>

Hosseini, H. G., Luo, D. and Reynolds, K. J. (2006) 'The Comparison of Different Feed Forward Neural Network Architectures for Ecg Signal Diagnosis.' *Medical Engineering & Physics* 28, (4) 372-378 available from <http://www.sciencedirect.com/science/article/pii/S1350453305001475>

Howard, A. (1994) *Elementary Linear Algebra*. 7 edn.: John Wiley & Sons

Hsu, C.-W., Chang, C.-C. and Lin, C.-J. (2010) *A Practical Guide to Support Vector Classification.* National Taiwan University. Available from <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>

Htc Corporation (2010) *Htc Hd2* [online] available from <http://www.htc.com/uk/product/hd2/overview.html> [December

Hu, Y.-H., Palreddy, S. and Tompkins, W. J. (1997) 'A Patient-Adaptable Ecg Beat Classifier Using a Mixture of Experts Approach.' *Biomedical Engineering, IEEE Transactions on* 44, (9) 891-900

Hu, Y. H., Palreddy, S. and Tompkins, W. J. (1997) 'A Patient-Adaptable Ecg Beat Classifier Using a Mixture of Experts Approach.' *IEEE Transactions on Biomedical Engineering* 44, (9) 891-900 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0031238509&partnerID=40&md5=a2ac4058ae3320a98c55a99603b9849a>

Hu, Y. H., Tompkins, W. J., Urrusti, J. L. and Afonso, V. X. (1994) 'Applications of Artificial Neural Networks for Ecg Signal Detection and Classification.' *Journal of Electrocardiology* 26, (SUPPL.) 66-73 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0028205396&partnerID=40&md5=f0ebec05d5298bff882268ff885fc89b>

Huang, H., Liu, J., Zhu, Q., Wang, R. and Hu, G. (2014) 'A New Hierarchical Method for Inter-Patient Heartbeat Classification Using Random Projections and Rr Intervals.' *BioMedical Engineering OnLine* 13, (1) 90 available from <http://www.biomedical-engineering-online.com/content/13/1/90>

Hughes, N. P., Tarassenko, L. and Roberts, S. J. (2004) 'Markov Models for Automated Ecg Interval Analysis.' *Advances in Neural Information Processing Systems (NIPS)* 16, 611-618

Hutchison, J. S., Ward, R. E., Lacroix, J., Hébert, P. C., Barnes, M. A., Bohn, D. J., Dirks, P. B., Doucette, S., Fergusson, D., Gottesman, R., Joffe, A. R., Kirpalani, H. M., Meyer, P. G., Morris, K. P., Moher, D., Singh, R. N. and Skippen, P. W. (2008) 'Hypothermia Therapy after Traumatic Brain Injury in Children.' *New England Journal of Medicine* 358, (23) 2447-2456 available from <http://www.nejm.org/doi/abs/10.1056/NEJMoa0706930>

Ibarz, A., Falcó, J., Vaquerizo, E., Lain, L., Artigas, J. and Roy, A. (2012) 'Monami: Mainstream on Ambient Intelligence. Scaled Field Trial Experience in a Spanish Geriatric Residence.' In *Ambient Assisted Living and Home Care.* vol. 7657 ed. by Bravo, J., Hervás, R. and Rodríguez, M.: Springer Berlin Heidelberg: 119-126. Available from <http://dx.doi.org/10.1007/978-3-642-35395-6_17>

Inan, O. T., Giovangrandi, L. and Kovacs, G. T. A. (2006) 'Robust Neural-Network-Based Classification of Premature Ventricular Contractions Using Wavelet Transform and Timing Interval Features.' *IEEE Transactions on Biomedical Engineering* 53, (12) 2507-2515 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-33845865323&partnerID=40&md5=17da9c724ba96a4d2ab5b43b1ea15324>

Intercure, I. (n.d.) *Resperate* [online] available from <http://www.resperate.co.uk/> [December 2014]

Istepanian, R. S. H., Jovanov, E. and Zhang, Y. T. (2004) 'Guest Editorial Introduction to the Special Section on M-Health: Beyond Seamless Mobility and Global Wireless Health-Care Connectivity.' *Information Technology in Biomedicine, IEEE Transactions on* 8, (4) 405-414

Istepanian, R. S. H., Laxminarayan, S. and Pattichis, C. S. (2006) *M-Health: Emerging Mobile Health Systems.* Springer US

Jane, R., Rix, H., Caminal, P. and Laguna, P. (1991) 'Alignment Methods for Averaging of High-Resolution Cardiac Signals: A Comparative Study of Performance.' *IEEE Trans Biomed Eng* 38, (6) 571-579

Jansen, J. O. and Cuthbertson, B. H. (2010) 'Detecting Critical Illness Outside the Icu: The Role of Track and Trigger Systems.' *Curr Opin Crit Care* 16, (3) 184-190

Javadi, M., Arani, S. a. a. A., Sajedin, A. and Ebrahimpour, R. (2013) 'Classification of Ecg Arrhythmia by a Modular Neural Network Based on Mixture of Experts and Negatively Correlated Learning.' *Biomedical Signal Processing and Control* 8, (3) 289-296 available from <http://www.sciencedirect.com/science/article/pii/S1746809412001127>

Jekova, I., Bortolan, G. and Christov, I. (2004) 'Pattern Recognition and Optimal Parameter Selection in Premature Ventricular Contraction Classification.' *Computers in Cardiology, 2004.* 19-22 Sept. 2004

Jiang, W. and Kong, S. G. (2007) 'Block-Based Neural Networks for Personalized Ecg Signal Classification.' *IEEE Transactions on Neural Networks* 18, (6) 1750-1761 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-36348956823&partnerID=40&md5=dfc6b1f9abdd866fcfb85d7a2dd5fb35>

Jones, W. D. (2006) 'Taking Body Temperature, inside Out.' *IEEE Spectrum* (43) 13-15

Joone (2007) *Java Object Oriented Neural Engine (Joone 2.0)* [online] available from <http://www.joone.org> [August 2010]

Jovanov, E., O'donnell Lords, A., Raskovic, D., Cox, P. G., Adhami, R. and Andrasik, F. (2003) 'Stress Monitoring Using a Distributed Wireless Intelligent Sensor System.' *Engineering in Medicine and Biology Magazine, IEEE* 22, (3) 49-55

Ka, A. K. (2012) 'Ecg Beat Classification Using Waveform Similarity and Rr Intervals ' *Journal of Medical and Biological Engineering* 32, (6) 417-422

Kadambe, S., Murray, R. and Boudreaux-Bartels, G. F. (1999) 'Wavelet Transform-Based Qrs Complex Detector.' *Biomedical Engineering, IEEE Transactions on* 46, (7) 838-848

Kamel Boulos, M. N., Lou, R. C., Anastasiou, A., Nugent, C. D., Alexandersson, J., Zimmermann, G., Cortes, U. and Casas, R. (2009) 'Connectivity for Healthcare and Well-Being Management: Examples from Six European Projects.' *Int J Environ Res Public Health* 6, (7) 1947-1971

Kampouraki, A., Manis, G. and Nikou, C. (2009) 'Heartbeat Time Series Classification with Support Vector Machines.' *Information Technology in Biomedicine, IEEE Transactions on* 13, (4) 512-518

Kannathal, N., Acharya, U. R., Ng, E. Y. K., Krishnan, S. M., Min, L. C. and Laxminarayan, S. (2006) 'Cardiac Health Diagnosis Using Data Fusion of Cardiovascular and Haemodynamic Signals.' *Computer Methods and Programs in Biomedicine* 82, (2) 87-96 available from <http://www.cmpbjournal.com/article/S0169-2607(06)00050-2/abstract> [2014/06/26]

Kanoun, K., Kaaniche, M. and Laprie, J. C. (1997) 'Qualitative and Quantitative Reliability Assessment.' *Software, IEEE* 14, (2) 77-87

Kaplan, R. M. (2001) 'Health Outcomes, Assessment Of.' In *International Encyclopedia of the Social & Behavioral Sciences.* Oxford: Pergamon: 6581-6586. Available from <http://www.sciencedirect.com/science/article/B7MRM-4MT09VJ-20M/2/a0464c261a5af5bef92246183b55a679>

Kaplan, R. M., Bush, J. W. and Berry, C. C. (1976) 'Health Status: Types of Validity and the Index of Well-Being.' *Health services research* 11, (4) 478–507

Karpagachelvi, S., Arthanari, M. and Sivakumar, M. (2010) 'Ecg Feature Extraction Techniques - a Survey Approach.' *(IJCSIS) International Journal of Computer Science and Information Security* 8, (1)

Kasper, D., Braunwald, E., Fauci, A., Hauser, S., Longo, D., Jameson, J. and Loscalzo, J. (ed.) (2008) *Harrison's Principles of Internal Medicine*. 17 edn. New York: McGraw-Hill Medical Publishing Division

Katsis, C. D., Goletsis, Y., Rigas, G. and Fotiadis, D. I. (2011) 'A Wearable System for the Affective Monitoring of Car Racing Drivers During Simulated Conditions.' *Transportation Research Part C: Emerging Technologies* 19, (3) 541-551 available from <http://www.sciencedirect.com/science/article/pii/S0968090X10001403>

Kazman, R., Abowd, G., Bass, L. and Clements, P. (1996) 'Scenario-Based Analysis of Software Architecture.' *Software, IEEE* 13, (6) 47-55

Kazman, R., Klein, M., Barbacci, M., Longstaff, T., Lipson, H. and Carriere, J. (1998) 'The Architecture Tradeoff Analysis Method.' *Engineering of Complex Computer Systems, 1998. ICECCS '98. Proceedings. Fourth IEEE International Conference on*. 10-14 Aug 1998

Keerthi, S. S. and Lin, C.-J. (2003) 'Asymptotic Behaviors of Support Vector Machines with Gaussian Kernel.' *Neural Comput.* 15, (7) 1667-1689

Keerthi, S. S., Shevade, S. K., Bhattacharyya, C. and Murthy, K. R. K. (2001) 'Improvements to Platt's Smo Algorithm for Svm Classifier Design.' *Neural Comput.* 13, (3) 637-649

Kendrick, A. (2008) 'Pulse Oximetry.' In *The Buyers' Guide to Respiratory Care Products.* European Respiratory Society: 190-214

Kenneth, E., Rajendra Acharya, U., Kannathal, N. and Lim Choo, M. (2005) 'Data Fusion of Multimodal Cardiovascular Signals.' *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. 2005

Kim, J., Min, S. D. and Lee, M. (2011) 'An Arrhythmia Classification Algorithm Using a Dedicated Wavelet Adapted to Different Subjects.' *BioMedical Engineering OnLine* 10, available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-79959550202&partnerID=40&md5=e6d7bbd866a3a354afaa4a8995efe98e>

Kimeldorf, G. S. and Wahba, G. (1970) 'A Correspondence between Bayesian Estimation on Stochastic Processes and Smoothing by Splines.' (2) 495-502 available from <http://projecteuclid.org/euclid.aoms/1177697089>

Kirsch, C., Mattingley-Scott, M., Muszynski, C., Schaefer, F. and Weiss, C. (2007) 'Monitoring Chronically Ill Patients Using Mobile Technologies.' *IBM Syst. J.* 46, (1) 85-93

Klabunde, R. E. (2005) *Cardiovascular Physiology Concepts*. Lippincott Williams & Wilkins

Kligfield, P., Gettes, L. S., Bailey, J. J., Childers, R., Deal, B. J., Hancock, E. W., Van Herpen, G., Kors, J. A., Macfarlane, P., Mirvis, D. M., Pahlm, O., Rautaharju, P. and Wagner, G. S. (2007) 'Recommendations for the Standardization and Interpretation of the Electrocardiogrampart I: The Electrocardiogram and Its Technology a Scientific Statement from the American Heart Association Electrocardiography and Arrhythmias

Committee, Council on Clinical Cardiology; the American College of Cardiology Foundation; and the Heart Rhythm Society Endorsed by the International Society for Computerized Electrocardiology.' *Journal of the American College of Cardiology* 49, (10) 1109-1127 available from <http://dx.doi.org/10.1016/j.jacc.2007.01.024>

Kluge, E.-H. W. (2011) 'Ethical and Legal Challenges for Health Telematics in a Global World: Telehealth and the Technological Imperative.' *International Journal of Medical Informatics* 80, (2) e1-e5 available from <http://www.sciencedirect.com/science/article/pii/S1386505610001735>

Knight, F., Schwirtz, A., Psomadelis, F., Baber, C., Bristow, W. and Arvanitis, N. (2005) 'The Design of the Sensvest.' *Personal Ubiquitous Comput.* 9, (1) 6-19

Knudsen, J. (2008) 'The Lightweight User Interface Toolkit (Lwuit): An Introduction.' *ORACLE Technical Articles and Tips* [online] [10 2010]

Ko, J., Lu, C., Srivastava, M. B., Stankovic, J. A., Terzis, A. and Welsh, M. (2010) 'Wireless Sensor Networks for Healthcare.' *Proceedings of the IEEE* 98, (11) 1947-1960

Kohler, B. U., Hennig, C. and Orglmeister, R. (2002) 'The Principles of Software Qrs Detection.' *Engineering in Medicine and Biology Magazine, IEEE* 21, (1) 42-57

Konstantas, D., Jones, V., Bults, R. and Herzog, R. (2002) *Mobihealth – Innovative 2.5 / 3g Mobile Services and Applications for Healthcare.* 11th IST Mobile and Wireless Telecommunications Summit. Thessaloniki, Greece

Korhonen, I., Parkka, J. and Van Gils, M. (2003) 'Health Monitoring in the Home of the Future.' *Engineering in Medicine and Biology Magazine, IEEE* 22, (3) 66-73

Kotas, M. (2004) 'Projective Filtering of Time-Aligned Ecg Beats.' *Biomedical Engineering, IEEE Transactions on* 51, (7) 1129-1139

Laerhoven, K. V., Lo, B. P., Ng, J. W., Thiemjarus, S., King, R., Kwan, S., Gellersen, H.-W., Sloman, M., Wells, O., Needham, P., Peters, N., Darzi, A., Toumazou, C. and Yang, G.-Z. (2004) 'Medical Healthcare Monitoring with Wearable and Implantable Sensors.' *Sixth International Conference on Ubiquitous Computing.* Held at Japan

Lagerholm, M., Peterson, C., Braccini, G., Edenbrandt, L. and Sornmo, L. (2000) 'Clustering Ecg Complexes Using Hermite Functions and Self-Organizing Maps.' *Biomedical Engineering, IEEE Transactions on* 47, (7) 838-848

Larson, R. (1978) 'Thirty Years of Research on the Subjective Well-Being of Older Americans.' *J Gerontol* 33, (1) 109-125 available from <http://geronj.oxfordjournals.org/cgi/content/abstract/33/1/109>

Lee, H. J., Lee, S. H., Ha, K. S., Jang, H. C., Chung, W. Y., Kim, J. Y., Chang, Y. S. and Yoo, D. H. (2009) 'Ubiquitous Healthcare Service Using Zigbee and Mobile Phone for Elderly Patients.' *International Journal of Medical Informatics* 78, (3) available from <http://www.sciencedirect.com/science/article/B6T7S-4TB0V91-1/2/74c4d80285003417ea92316705997dd5>

Lee, J.-W. and Lee, G.-K. (2005) 'Design of an Adaptive Filter with a Dynamic Structure for Ecg Signal Processing.' *International Journal of Control, Automation, and Systems* 3, (1) 137-142

Lee, J., Jeong, K., Yoon, J. and Lee, M. (1996) 'A Simple Real-Time Qrs Detection Algorithm.' *vol. 4, Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine. Proceedings of the 18th Annual International Conference of the IEEE*. 31 Oct-3 Nov 1996

Leeds Teaching Hospitals Nhs Trust (2010) *Adult Modified Early Warning Scores (Mews) and Graded Response Protocol*. Available from <http://www.nhstaps.org/site_media/uploads/blog/mews_protocol.pdf>

Leon, E., Clarke, G., Callaghan, V. and Sepulveda, F. (2007) 'A User-Independent Real-Time Emotion Recognition System for Software Agents in Domestic Environments.' *Engineering Applications of Artificial Intelligence* 20, (3) 337-345 available from <http://www.sciencedirect.com/science/article/B6V2M-4KGPNC7-2/2/4e4d2954adf170f9c8ac33727b80d77b>

Leuner, J. (2004) *Jazzlib: A Pure Java Implementation of the Java.Util.Zip Library* [online] available from <http://jazzlib.sourceforge.net/> [September 2011]

Levis, P. (2006) *Tinyos 2.0 Overview* [online] available from <http://www.tinyos.net/tinyos-2.x/doc/html/overview.html>

Levis, P. (2006) 'Tinyos Programming.' [online]. Available from <http://csl.stanford.edu/~pal/pubs/tinyos-programming.pdf>

Lewandowski, J., Arochena, H. E., Naguib, R. N. G. and Chao, K. (2012) 'A Simple Real-Time Qrs Detection Algorithm Utilizing Curve-Length Concept with Combined Adaptive Threshold for Electrocardiogram Signal Classification.' *TENCON 2012 - 2012 IEEE Region 10 Conference*. 19-22 Nov. 2012

Lifesync Corporation (2009) *Lifesync Wireless Ecg System* [online] available from <LifeSync Corporation> [September 2009]

Lifewatch Ag (n.d.) *Wireless Blood Pressure Monitors* [online] available from <http://www.lifewatch.com/siteFiles/1/538/6372.asp> [September 2009]

Lin, C.-H. (2008) 'Frequency-Domain Features for Ecg Beat Discrimination Using Grey Relational Analysis-Based Classifier.' *Computers & Mathematics with Applications* 55,

(4) 680-690 available from
<http://www.sciencedirect.com/science/article/pii/S0898122107005019>

Lin, H.-T. and Lin, C.-J. (2003) *A Study on Sigmoid Kernels for Svm and the Training of Non-Psd Kernels by Smo-Type Methods.* National Taiwan University. Available from <http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf>

Linh, T. H., Osowski, S. and Stodolski, M. (2003) 'On-Line Heart Beat Recognition Using Hermite Polynomials and Neuro-Fuzzy Network.' *IEEE Transactions on Instrumentation and Measurement* 52, (4) 1224-1231 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0141919673&partnerID=40&md5=cc5c88ff2dcdc59467f49e234d0ccbcd>

Lisetti, C. L. and Nasoz, F. (2004) 'Using Noninvasive Wearable Computers to Recognize Human Emotions from Physiological Signals.' *EURASIP J. Appl. Signal Process.* 2004, 1672-1687

Liu, A. and Gorton, I. (2003) 'Accelerating Cots Middleware Acquisition: The I-Mate Process.' *Software, IEEE* 20, (2) 72-79

Liu, Y., Gorton, I., Bass, L., Hoang, C. and Abanmi, S. (2006) 'Mems: A Method for Evaluating Middleware Architectures.' In *Quality of Software Architectures.* vol. 4214 ed. by Hofmeister, C., Crnkovic, I. and Reussner, R.: Springer Berlin Heidelberg: 9-26. Available from <http://dx.doi.org/10.1007/11921998_6>

Llamedo, M. and Martinez, J. P. (2011) 'Heartbeat Classification Using Feature Selection Driven by Database Generalization Criteria.' *Biomedical Engineering, IEEE Transactions on* 58, (3) 616-625

Llamedo, M. and Martínez, J. P. (2011) 'Heartbeat Classification Using Feature Selection Driven by Database Generalization Criteria.' *IEEE Transactions on Biomedical Engineering* 58, (3 PART 1) 616-625 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-79952156441&partnerID=40&md5=75aeacbfe14310620471c61ba23c9533>

Lymberis, A. (2003) 'Smart Wearable Systems for Personalised Health Management: Current R&D and Future Challenges.' *vol. 4, Engineering in Medicine and Biology Society, 2003. Proceedings of the 25th Annual International Conference of the IEEE.*

Lymberis, A. and Dittmar, A. (2007) 'Advanced Wearable Health Systems and Applications - Research and Development Efforts in the European Union.' *Engineering in Medicine and Biology Magazine, IEEE* 26, (3) 29-33

Lymberis, A. and Olsson, S. (2003) 'Intelligent Biomedical Clothing for Personal Health and Disease Management: State of the Art and Future Vision.' *Telemed J E Health* 9, (4) 379-386

Ma, Z. (2011) 'An Electronic Second Skin.' *Science* 333, (6044) 830-831 available from
<http://www.sciencemag.org/content/333/6044/830.short>

Mackowiak, P. A., Wasserman, S. S. and Levine, M. M. (1992) 'A Critical Appraisal of 98.6â°F,
the Upper Limit of the Normal Body Temperature, and Other Legacies of Carl Reinhold
August Wunderlich.' *JAMA: The Journal of the American Medical Association* 268, (12)
1578-1580 available from <http://jama.ama-assn.org/content/268/12/1578.abstract>

Majeed, B. A. and Brown, S. J. (2006) 'Developing a Well-Being Monitoring System—Modeling
and Data Analysis Techniques.' *Applied Soft Computing* 6, 384–393

Manikandan, M. S. and Soman, K. P. (2012) 'A Novel Method for Detecting R-Peaks in
Electrocardiogram (Ecg) Signal.' *Biomedical Signal Processing and Control* 7, (2) 118-
128 available from
<http://www.sciencedirect.com/science/article/pii/S1746809411000292>

Mar, T., Zaunseder, S., Marti, X, Nez, J. P., Llamedo, M. and Poll, R. (2011) 'Optimization of
Ecg Classification by Means of Feature Selection.' *Biomedical Engineering, IEEE
Transactions on* 58, (8) 2168-2177

Mare, S. and Kotz, D. (2010) 'Is Bluetooth the Right Technology for Mhealth?' *USENIX
Workshop on Health Security and Privacy*. Available from <http://sharps.org/wp-
content/uploads/MARE-HEALTHSEC10.pdf>

Marks, M., South, M. and Carter, B. (1995) 'Measurement of Respiratory Rate and Timing Using
a Nasal Thermocouple.' *Journal of Clinical Monitoring and Computing* 11, (3) 159-164
available from <http://dx.doi.org/10.1007/BF01617716>

Marrone, P. (2007) *Joone: The Complete Guide* [online]. Available from <http://www.joone.org>

Martin-Lesende, I., Orruno, E., Bilbao, A., Vergara, I., Cairo, M. C., Bayon, J. C., Reviriego, E.,
Romo, M. I., Larranaga, J., Asua, J., Abad, R. and Recalde, E. (2013) 'Impact of
Telemonitoring Home Care Patients with Heart Failure or Chronic Lung Disease from
Primary Care on Healthcare Resource Use (the Telbil Study Randomised Controlled
Trial).' *BMC Health Services Research* 13, (1) 118 available from
<http://www.biomedcentral.com/1472-6963/13/118>

Martin, A., Lassman, D., Whittle, L., Catlin, A. and Team, T. N. H. E. A. (2011) 'Recession
Contributes to Slowest Annual Rate of Increase in Health Spending in Five Decades.'
Health Affairs 30, (1) 11-22 available from
<http://content.healthaffairs.org/content/30/1/11.abstract>

Martínez, J. P., Almeida, R., Olmos, S., Rocha, A. P. and Laguna, P. (2004) 'A Wavelet-Based
Ecg Delineator Evaluation on Standard Databases.' *IEEE Transactions on Biomedical
Engineering* 51, (4) 570-581 available from
<http://www.scopus.com/inward/record.url?eid=2-s2.0-
12144285744&partnerID=40&md5=2a39c2a2d988be20f261333221e652dd>

Mattsson, M., Grahn, H. and Mårtensson, F. (2006) 'Software Architecture Evaluation Methods for Performance, Maintainability, Testability, and Portability.' *Second International Conference on the Quality of Software Architectures.*

Mcgrath, M. J. and Dishongh, T. J. (2009) 'A Common Personal Health Research Platform—Shimmer and Biomobius.' *Intel Technology Journal* 13, (3) 122-147

Medapps Ltd. (n.d.) [online] available from <www.medapps.com > [August 2009]

Meddins, B. (2000) '5 - the Design of Fir Filters.' In *Introduction to Digital Signal Processing.* ed. by Meddins, B. Oxford: Newnes: 102-136. Available from <http://www.sciencedirect.com/science/article/pii/B9780750650489500076>

Medilog, S. (2010) *Schiller Medilog Holter* [online] available from <http://www.medilogholter.com.au/> [December

Mele, P. (2008) 'Improving Electrocardiogram Interpretation in the Clinical Setting.' *Journal of Electrocardiology* 41, (5) 438-439 available from <http://www.sciencedirect.com/science/article/pii/S0022073608001453> [2008/10//]

Melgani, F. and Bazi, Y. (2008) 'Classification of Electrocardiogram Signals with Support Vector Machines and Particle Swarm Optimization.' *Information Technology in Biomedicine, IEEE Transactions on* 12, (5) 667-677

Miklos, M., Kusy, B., Simon, G. and Ledeczi, A. (2004) *The Flooding Time Synchronization Protocol.* Proceedings of the 2nd international conference on Embedded networked sensor systems. Baltimore, MD, USA: ACM

Milivojević, Z. (2009) *Digital Filter Design* [online]: mikroElektronika. Available from <http://www.mikroe.com/products/view/268/digital-filter-design/>

Miroslav, K., Fedor, L. and Gabriel, V. (2012) 'Multi-Platform Telemedicine System for Patient Health Monitoring.' *Biomedical and Health Informatics (BHI), 2012 IEEE-EMBS International Conference on.* 5-7 Jan. 2012

Moavenian, M. and Khorrami, H. (2010) 'A Qualitative Comparison of Artificial Neural Networks and Support Vector Machines in Ecg Arrhythmias Classification.' *Expert Systems with Applications* 37, (4) 3088-3093 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-71349087690&partnerID=40&md5=ee4084b18d8bfa096bdf0b734447bde5>

Mobhealth (n.d.) [online] available from <http://sourceforge.net/projects/mobhealth/> [August 2009]

Mobihealth B.V. (2009) [online] available from <http://www.mobihealth.com> [August 2009]

Moody, G., Muldrow, W. and Mark, R. (1984) 'A Noise Stress Test for Arrhythmia Detectors.'
Computers in Cardiology 11, 381-384

Moody, G. B. (1997) *Mit-Bih Arrhythmia Database Directory* [online] available from
<http://www.physionet.org/physiobank/database/html/mitdbdir/mitdbdir.htm> [04-04-
2014

Moody, G. B. (2014) *Wfdb Applications Guide* [online]: Harvard-MIT Division of Health Sciences
and Technology. Available from <http://www.physionet.org/physiotools/wag/wag.pdf>

Moody, G. B. and Mark, R. G. (1996) 'A Database to Support Development and Evaluation of
Intelligent Intensive Care Monitoring.' *Computers in Cardiology, 1996.* 8-11 Sept. 1996

Moody, G. B. and Mark, R. G. (2001) 'The Impact of the Mit-Bih Arrhythmia Database.'
Engineering in Medicine and Biology Magazine, IEEE 20, (3) 45-50

Morgan, D. R. and Zierdt, M. G. (2009) 'Novel Signal Processing Techniques for Doppler Radar
Cardiopulmonary Sensing.' *Signal Processing* 89, (1) 45-66 available from
<http://www.sciencedirect.com/science/article/pii/S0165168408002211>

Morgan, R., Williams, F. and Wright, M. (1997) 'An Early Warning Scoring System for Detecting
Developing Critical Illness.' *Clin Intens Care* 8, 100

Morgan, R. J. M. and Wright, M. M. (2007) 'In Defence of Early Warning Scores.' *British Journal
of Anaesthesia* 99, (5) 747-748 available from
<http://bja.oxfordjournals.org/content/99/5/747.short>

Müller, S. (2002) *Jopenchart Library and Toolkit* [online] [10 2010]

Mundt, C. W., Montgomery, K. N., Udoh, U. E., Barker, V. N., Thonier, G. C., Tellier, A. M.,
Ricks, R. D., Darling, B. B., Cagle, Y. D., Cabrol, N. A., Ruoss, S. J., Swain, J. L.,
Hines, J. W. and Kovacs, G. T. A. (2005) 'A Multiparameter Wearable Physiologic
Monitoring System for Space and Terrestrial Applications.' *Information Technology in
Biomedicine, IEEE Transactions on* 9, (3) 382-391

Mytech Technology Ltd. (n.d.) [online] available from
<http://www.allproducts.com/communication/mytech/series2.html> [June 2009]

Nasa (n.d.) *Lifeguard Project* [online] available from <http://lifeguard.stanford.edu/> [2009
August]

Nasoz, F., Alvarez, K., Lisetti, C. L. and Finkelstein, N. (2004) 'Emotion Recognition from
Physiological Signals Using Wireless Sensors for Presence Technologies.' *Cognition,
Technology & Work* 6, (1) 4-14 available from

<http://search.ebscohost.com/login.aspx?direct=true&db=a9h&AN=12268347&site=ehost-live>

Nikolaev, N., Nikolov, Z., Gotchev, A. and Egiazarian, K. (2000) 'Wavelet Domain Wiener Filtering for Ecg Denoising Using Improved Signal Estimate.' *vol. 6, Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference on.* 2000

Nokia Corporation (2006) *Jsr 256: Mobile Sensor Api.* Available from <http://jcp.org/en/jsr/detail?id=256>

Nonin (2014) *Nonin Go2 Achieve 9570 Finger Pulse Oximeter* [online] available from <http://www.nonin.com/Finger-Pulse-Oximeter/Nonin-GO2-Achieve> [December 2014]

Nonin Medical Inc (2010) *Reusable Purelight® Oximetry Flex Sensors* [online] available from <http://www.nonin.com/PulseOximetry/Sensors/FlexSensors> [December 2010]

Oracle Corporation (2010) *Java 2 Software the Platforms* [online] available from <http://java.sun.com/java2/>

Orfanidis, S. J. (1995) *Introduction to Signal Processing.* Prentice-Hall, Inc.

Osowski, S. and Linh, T. H. (2001) 'Ecg Beat Recognition Using Fuzzy Hybrid Neural Network.' *IEEE Transactions on Biomedical Engineering* 48, (11) 1265-1271 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0034774017&partnerID=40&md5=f0a23a71231a0aafeaeb64bc563b0fdb>

Otto, C., Milenkovic, A., Sanders, C. and E., J. (2006) 'System Architecture of a Wireless Body Area Sensor Network for Ubiquitous Health Monitoring.' *Journal of Mobile Multimedia* 1, (4) 307-326

Özbay, Y. (2009) 'A New Approach to Detection of Ecg Arrhythmias: Complex Discrete Wavelet Transform Based Complex Valued Artificial Neural Network.' *Journal of Medical Systems* 33, (6) 435-445 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-70350228556&partnerID=40&md5=b8a49146be2e6d5494e828400a03e6fe>

Özbay, Y., Ceylan, R. and Karlik, B. (2006) 'A Fuzzy Clustering Neural Network Architecture for Classification of Ecg Arrhythmias.' *Computers in Biology and Medicine* 36, (4) 376-388 available from <http://www.sciencedirect.com/science/article/pii/S0010482505000417>

Özbay, Y., Ceylan, R. and Karlik, B. (2011) 'Integration of Type-2 Fuzzy Clustering and Wavelet Transform in a Neural Network Based Ecg Classifier.' *Expert Systems with Applications* 38, (1) 1004-1010 available from <http://www.sciencedirect.com/science/article/pii/S0957417410007463>

Özbay, Y. and Tezel, G. (2010) 'A New Method for Classification of Ecg Arrhythmias Using Neural Network with Adaptive Activation Function.' *Digital Signal Processing* 20, (4) 1040-1049 available from <http://www.sciencedirect.com/science/article/pii/S1051200409001948>

Pan, G. and Wang, L. (2012) 'Swallowable Wireless Capsule Endoscopy: Progress and Technical Challenges.' *Gastroenterology Research and Practice* 2012, 9 available from <http://dx.doi.org/10.1155/2012/841691>

Pan, J. and Tompkins, W. J. (1985) 'A Real-Time Qrs Detection Algorithm.' *IEEE Transactions on Biomedical Engineering* 32, (3) 230-236 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0021892137&partnerID=40&md5=8ff37dc586908bd463eea3cfcdfae3a6>

Pandian, P. S., Mohanavelu, K., Safeer, K. P., Kotresh, T. M., Shakunthala, D. T., Gopal, P. and Padaki, V. C. (2008) 'Smart Vest: Wearable Multi-Parameter Remote Physiological Monitoring System.' *Medical Engineering & Physics* 30, (4) 466-477 available from <http://www.sciencedirect.com/science/article/pii/S1350453307000975>

Paoletti, M. and Marchesi, C. (2006) 'Discovering Dangerous Patterns in Long-Term Ambulatory Ecg Recordings Using a Fast Qrs Detection Algorithm and Explorative Data Analysis.' *Computer Methods and Programs in Biomedicine* 82, (1) 20-30 available from <http://linkinghub.elsevier.com/retrieve/pii/S016926070600023X?showall=true>

Parker, S., Nussbaum, G., Sonntag, H., Pühretmair, F., Williams, V., Mccrindle, R., Victor, C., Oliver, D., Maguire, M., Mayer, P., Edelmayer, G. and Panek, P. (2008) 'Enable – a View on User's Needs.' In *Computers Helping People with Special Needs.* vol. 5105 ed. by Miesenberger, K., Klaus, J., Zagler, W. and Karshmer, A.: Springer Berlin Heidelberg: 1016-1023. Available from <http://dx.doi.org/10.1007/978-3-540-70540-6_152>

Patel, S., Park, H., Bonato, P., Chan, L. and Rodgers, M. (2012) 'A Review of Wearable Sensors and Systems with Application in Rehabilitation.' *Journal of NeuroEngineering and Rehabilitation* 9, (1) 21 available from <http://www.jneuroengrehab.com/content/9/1/21>

Patterson, C., Maclean, F., Bell, C., Mukherjee, E., Bryan, L., Woodcock, T. and Bell, D. (2011) 'Early Warning Systems in the Uk: Variation in Content and Implementation Strategy Has Implications for a Nhs Early Warning System.' *Clin Med* 11, (5) 424-427

Phoneme (n.d.) [online] available from <https://phoneme.dev.java.net/> [August 2010]

Platt, J. C. (1999) 'Fast Training of Support Vector Machines Using Sequential Minimal Optimization.' In *Advances in Kernel Methods—Support Vector Learning.* ed. by Sch¨olkopf, D., Burges, C. J. C. and Smola, A. J. Cambridge: MIT Press: 185–208

Polar Ltd (n.d.) [online] available from <http://www.polar.fi/en/products/, > [21 August 2009]

Polisena, J., Tran, K., Cimon, K., Hutton, B., Mcgill, S., Palmer, K. and Scott, R. E. (2010) 'Home Telemonitoring for Congestive Heart Failure: A Systematic Review and Meta-Analysis.' *Journal of Telemedicine and Telecare* 16, (2) 68-76 available from <http://jtt.sagepub.com/content/16/2/68.abstract>

Polymap Wirelles (2008) *Polytel® System* [online] available from < http://www.polymapwireless.com/> [August 2009]

Prytherch, D. R., Smith, G. B., Schmidt, P. E. and Featherstone, P. I. (2010) 'Views—Towards a National Early Warning Score for Detecting Adult Inpatient Deterioration.' *Resuscitation* 81, (8) 932-937 available from <http://www.sciencedirect.com/science/article/pii/S030095721000242X>

Qureshi, H. (2009) *Design of a Wireless Pulse Oximeter for Use in a Clinical Diagnostic System*. Unpublished thesis, McMaster University

Raatikainen, M. J. P., Uusimaa, P., Van Ginneken, M. M. E., Janssen, J. P. G. and Linnaluoto, M. (2008) 'Remote Monitoring of Implantable Cardioverter Defibrillator Patients: A Safe, Time-Saving, and Cost-Effective Means for Follow-Up.' *Europace* 10, (10) 1145-1151 available from <http://europace.oxfordjournals.org/cgi/content/abstract/10/10/1145>

Reussner, R. H., Schmidt, H. W. and Poernomo, I. H. (2003) 'Reliability Prediction for Component-Based Software Architectures.' *Journal of Systems and Software* 66, (3) 241-252 available from <http://www.sciencedirect.com/science/article/pii/S0164121202000808>

Rimet, Y., Brusquet, Y., Ronayette, D., Dageville, C., Lubrano, M., Mallet, E., Rambaud, C., Terlaud, C., Silve, J., Lerda, O., Netchiporouk, L. I. and Weber, J. L. (2007) 'Surveillance of Infants at Risk of Apparent Life Threatening Events (Alte) with the Bba Bootee: A Wearable Multiparameter Monitor.' *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*. 22-26 Aug. 2007

Robinson, S., Stroetmann, K. A. and Stroetmann, V. N. (2004) 'Tele-Homecare for Chronically Ill Persons: Pilot Trials, Medical Outcomes and Future Perspectives.' *Stud Health Technol Inform* 103, 197-205

Rodrigues, J., Caldeira, J. O. and Vaidya, B. (2009) 'A Novel Intra-Body Sensor for Vaginal Temperature Monitoring.' *Sensors* 9, (4) 2797-2808 available from <http://www.mdpi.com/1424-8220/9/4/2797/>

Rojas, S. V. and Gagnon, M. P. (2008) 'A Systematic Review of the Key Indicators for Assessing Telehomecare Cost-Effectiveness.' *Telemed J E Health* 14, (9) 896-904

Romero, I., Geng, D. and Berset, T. (2012) 'Adaptive Filtering in Ecg Denoising: A Comparative Study.' *Computing in Cardiology (CinC), 2012*. 9-12 Sept. 2012

Roving Networks (2010) *Bluetooth Module Rn-42* [online] available from
<http://www.rovingnetworks.com/documents/rn-42-ds.pdf> [December

Royal College of Physicians (2012) *National Early Warning Score (News): Standardising the Assessment of Acute-Illness Severity in the Nhs.* London: RCP

Rusch, T. L., Sankar, R. and Scharf, J. E. (1996) 'Signal Processing Methods for Pulse Oximetry.' *Computers in Biology and Medicine* 26, (2) 143-159 available from
<http://www.sciencedirect.com/science/article/B6T5N-3VXH45H-5/2/1d315f07717af59211a8ca76b140f911>

S.L.P. Inc. (2010) *Piezo Crystal Respiratory Effort Sensors* [online] available from
<http://sleepsense.com/home.html> [December 2010]

Saalasti, S. (2003) *Neural Networks for Heart Rate Time Series Analysis.* Unpublished: University of Jyväskylä, Finland. Available from
<https://jyx.jyu.fi/dspace/handle/123456789/13267>

Sahambi, J. S., Tandon, S. N. and Bhatt, R. K. P. (1997) 'Using Wavelet Transforms for Ecg Characterization.' *IEEE Engineering in Medicine and Biology Magazine* 16, (1) 77-83 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-0031013508&partnerID=40&md5=7edb1fcf79d63f369fd3da71d8367e7a>

Sameni, R., Shamsollahi, M. B., Jutten, C. and Clifford, G. D. (2007) 'A Nonlinear Bayesian Filtering Framework for Ecg Denoising.' *IEEE Trans Biomed Eng* 54, (12) 2172-2185

Samsung Ltd (2008) *Adidas Micoach* [online] available from
<http://www.adidas.com/uk/miCoach/> [21 August 2009]

Saritha, C., Sukanya, V. and Murthy, Y. N. (2008) 'Ecg Signal Analysis Using Wavelet Transforms.' *Bulg. J. Phys* 35, (1) 68-77

Sayadi, O., Shamsollahi, M. B. and Clifford, G. D. (2010) 'Robust Detection of Premature Ventricular Contractions Using a Wave-Based Bayesian Framework.' *Biomedical Engineering, IEEE Transactions on* 57, (2) 353-362

Sayed, A. H. and Nascimento, V. H. (2005) 'Energy Conservation and the Learning Ability of Lms Adaptive Filters.' In *Least-Mean-Square Adaptive Filters.* John Wiley & Sons, Inc.: 79-104. Available from <http://dx.doi.org/10.1002/0471461288.ch3>

Schatzmann, J. and Ghanem, M. (2003) *Using Self-Organizing Maps to Visualize Clusters and Trends in Multidimensional Datasets.* London: Department of Computing Data Mining Group, Imperial College

Schein, M. H., Gavish, B., Herz, M., Rosner-Kahana, D., Naveh, P., Knishkowy, B., Zlotnikov, E., Ben-Zvi, N. and Melmed, R. N. (2001) 'Treating Hypertension with a Device That Slows and Regularises Breathing: A Randomised, Double-Blind Controlled Study.' *J Hum Hypertens* 15, (4) 271-278

Serafini, T., Zanghirati, G. and Zanni, L. (2005) 'Gradient Projection Methods for Quadratic Programs and Applications in Training Support Vector Machines.' *Optimization Methods and Software* 20, (2-3) 353-378 available from <http://dx.doi.org/10.1080/10556780512331318182> [2014/06/09]

Seto, E. (2008) 'Cost Comparison between Telemonitoring and Usual Care of Heart Failure: A Systematic Review.' *Telemed J E Health* 14, (7) 679-686

Shah-Hosseini, H. and Safabakhsh, R. (2000) 'Tasom: The Time Adaptive Self-Organizing Map.' *Information Technology: Coding and Computing, 2000. Proceedings. International Conference on*. 2000

Shelley, K. H. (2007) 'Photoplethysmography: Beyond the Calculation of Arterial Oxygen Saturation and Heart Rate.' *Anesthesia & Analgesia* 105, (6S Suppl) S31-S36 available from <http://www.anesthesia-analgesia.org/content/105/6S_Suppl/S31.abstract>

Shen, C.-P., Kao, W.-C., Yang, Y.-Y., Hsu, M.-C., Wu, Y.-T. and Lai, F. (2012) 'Detection of Cardiac Arrhythmia in Electrocardiograms Using Adaptive Feature Extraction and Modified Support Vector Machines.' *Expert Systems with Applications* 39, (9) 7845-7852 available from <http://www.sciencedirect.com/science/article/pii/S0957417412001066>

Shephard, R. J. (2003) 'Limits to the Measurement of Habitual Physical Activity by Questionnaires.' *British Journal of Sports Medicine* 37, (3) 197-206 available from <http://bjsm.bmj.com/content/37/3/197.abstract>

Shimmer Research (2010) [online] available from <http://www.shimmer-research.com/> [August 2010]

Shnayder, V., Chen, B., Lornicz, K., Fulford-Jones, T. R. F. and Welsh, M. (2005) *Sensor Networks for Medical Care.* Technical Report. Harvard University

Sixsmith, A., Meuller, S., Lull, F., Klein, M., Bierhoff, I., Delaney, S. and Savage, R. (2009) 'Soprano – an Ambient Assisted Living System for Supporting Older People at Home.' In *Ambient Assistive Health and Wellness Management in the Heart of the City*. vol. 5597 ed. by Mokhtari, M., Khalil, I., Bauchet, J., Zhang, D. and Nugent, C.: Springer Berlin Heidelberg: 233-236. Available from <http://dx.doi.org/10.1007/978-3-642-02868-7_30>

Smith, G. B., Prytherch, D. R., Schmidt, P., Featherstone, P. I., Knight, D., Clements, G. and Mohammed, M. A. (2006) 'Hospital-Wide Physiological Surveillance-a New Approach to the Early Identification and Management of the Sick Patient.' *Resuscitation* 71, (1) 19-

28 available from <http://www.sciencedirect.com/science/article/B6T19-4KSD8BH-7/2/3724f13576a48bb0615c25c55f76b90a>

Smith, G. B., Prytherch, D. R., Schmidt, P. E. and Featherstone, P. I. (2008a) 'Review and Performance Evaluation of Aggregate Weighted 'Track and Trigger' Systems.' *Resuscitation* 77, (2) 170-179

Smith, G. B., Prytherch, D. R., Schmidt, P. E., Featherstone, P. I. and Higgins, B. (2008b) 'A Review, and Performance Evaluation, of Single-Parameter "Track and Trigger" Systems.' *Resuscitation* 79, (1) 11-21

Smith, L. I. (2002) 'A Tutorial on Principal Components Analysis.' *Cornell University, USA* 51, 52

Sneha, S. and Varshney, U. (2009) 'Enabling Ubiquitous Patient Monitoring: Model, Decision Protocols, Opportunities and Challenges.' *Decision Support Systems* 46, (3) 606-619 available from <http://www.sciencedirect.com/science/article/pii/S0167923608002030>

Soler, V., Peñalver, A., Zuffanelli, S., Roig, J. and Aguiló, J. (2010) 'Domotic Hardware Infrastructure in Persona Project.' In *Ambient Intelligence and Future Trends-International Symposium on Ambient Intelligence (Isami 2010).* vol. 72 ed. by Augusto, J., Corchado, J., Novais, P. and Analide, C.: Springer Berlin Heidelberg: 149-155. Available from <http://dx.doi.org/10.1007/978-3-642-13268-1_18>

Steele, R., Lo, A., Secombe, C. and Wong, Y. K. (2009) 'Elderly Persons' Perception and Acceptance of Using Wireless Sensor Networks to Assist Healthcare.' *International Journal of Medical Informatics* 78, (12) 788-801 available from <http://www.sciencedirect.com/science/article/pii/S1386505609001178>

Suarez, K. V., Silva, J. C., Berthoumieu, Y., Gomis, P. and Najim, M. (2007) 'Ecg Beat Detection Using a Geometrical Matching Approach.' *Biomedical Engineering, IEEE Transactions on* 54, (4) 641-650

Subbe, C. P., Davies, R. G., Williams, E., Rutherford, P. and Gemmell, L. 'Effect of Introducing the Modified Early Warning Score on Clinical Outcomes, Cardio-Pulmonary Arrests and Intensive Care Utilisation in Acute Medical Admissions.' (0003-2409 (Print))

Subbe, C. P., Kruger, M., Rutherford, P. and Gemmel, L. (2001) 'Validation of a Modified Early Warning Score in Medical Admissions.' *QJM* 94, (10) 521-526 available from <http://qjmed.oxfordjournals.org/cgi/content/abstract/94/10/521>

Sund-Levander, M., Forsberg, C. and Wahren, L. K. (2002) 'Normal Oral, Rectal, Tympanic and Axillary Body Temperature in Adult Men and Women: A Systematic Literature Review.' *Scandinavian Journal of Caring Sciences* 16, (2) 122-128 available from <http://dx.doi.org/10.1046/j.1471-6712.2002.00069.x>

Sung, M., Marci, C. and Pentland, A. (2005) 'Wearable Feedback Systems for Rehabilitation.' *J Neuroeng Rehabil* 2, 17

Swan, A. and Sandilands, M. (1995) *Introduction to Geological Data Analysis*. Wiley

Swiatek, P., Stelmach, P., Prusiewicz, A. and Juszczyszyn, K. (2012) 'Service Composition in Knowledge-Based Soa Systems.' *New Generation Computing* 30, (2-3) 165-188 available from <http://dx.doi.org/10.1007/s00354-012-0204-x>

Tadejko, P. and Rakowski, W. (2007) 'Mathematical Morphology Based Ecg Feature Extraction for the Purpose of Heartbeat Classification.' *Computer Information Systems and Industrial Management Applications, 2007. CISIM '07. 6th International Conference on*. 28-30 June 2007

Tarassenko, L., Hann, A., Patterson, A., Braithwaite, E., Davidson, K., Barber, V. and Young, D. (2005) 'Biosign&Trade; : Multi-Parameter Monitoring for Early Warning of Patient Deterioration.' *Medical Applications of Signal Processing, 2005. The 3rd IEE International Seminar on (Ref. No. 2005-1119)*. 3-4 Nov. 2005

Tay, F. E. H., Guo, D. G., Xu, L., Nyan, M. N. and Yap, K. L. (2009) 'Memswear-Biomonitoring System for Remote Vital Signs Monitoring.' *Journal of the Franklin Institute* 346, (6) 531-542 available from <http://www.sciencedirect.com/science/article/B6V04-4VRX69K-1/2/d360837ac248e0b0a8af67443f32e50f>

Tazari, M.-R. (2010) 'An Open Distributed Framework for Adaptive User Interaction in Ambient Intelligence.' In *Ambient Intelligence.* vol. 6439 ed. by de Ruyter, B., Wichert, R., Keyson, D., Markopoulos, P., Streitz, N., Divitini, M., Georgantas, N. and Mana Gomez, A.: Springer Berlin Heidelberg: 227-238. Available from <http://dx.doi.org/10.1007/978-3-642-16917-5_23>

Texas Instruments (2006) *Msp430x1xx Family: User's Guide* [online]: Texas Instruments Incorporated

Thakor, N. V. and Yi-Sheng, Z. (1991) 'Applications of Adaptive Filtering to Ecg Analysis: Noise Cancellation and Arrhythmia Detection.' *Biomedical Engineering, IEEE Transactions on* 38, (8) 785-794

Thaler, M. S. (2007) *Only Ekg Book You'll Ever Need*. 5th edn.: Lippincott Williams & Wilkins

The National Coalition on Health Care (2009) 'Health Insurance Costs.' [online]. Available from <http://www.nchc.org/facts/cost.shtml> [August 2009]

Thoraval, L., Carrault, G., Schleich, J., Summers, R., Van De Velde, M. and Diaz, J. (1997) 'Data Fusion of Electrophysiological and Haemodynamic Signals for Ventricular Rhythm Tracking.' *Engineering in Medicine and Biology Magazine, IEEE* 16, (6) 48-55

Tia, G., Pesto, C., Selavo, L., Yin, C., Jeong Gil, K., Jong Hyun, L., Terzis, A., Watt, A., Jeng, J., Bor-Rong, C., Lorincz, K. and Welsh, M. (2008) 'Wireless Medical Sensor Networks in

Emergency Response: Implementation and Pilot Results.' *Technologies for Homeland Security, 2008 IEEE Conference on.*

Tinyos (n.d.) [online] available from <http://www.tinyos.net/> [August 2009]

Townsend, N. (2001) *Pulse Oximetry* [online] available from <http://www.robots.ox.ac.uk/~neil/teaching/lectures/med_elec/notes6.pdf>

Trafton, A. (2009) 'Wearable Blood Pressure Sensor Offers 24/7 Continuous Monitoring ' *MIT news* April 8: 4. Available from <http://web.mit.edu/newsoffice/2009/blood-pressure-tt0408.html>

Trossen, D. and Pavel, D. (2005) 'Building a Ubiquitous Platform for Remote Sensing Using Smartphones.' *Mobile and Ubiquitous Systems: Networking and Services, 2005. MobiQuitous 2005. The Second Annual International Conference on.* 17-21 July 2005

Trossen, D. and Pavel, D. (2006) *N-Rsa High-Level System Architecture.* Helsinki: Nokia Research Center

Tu, J. V. (1996) 'Advantages and Disadvantages of Using Artificial Neural Networks Versus Logistic Regression for Predicting Medical Outcomes.' *Journal of Clinical Epidemiology* 49, (11) 1225-1231 available from <http://www.sciencedirect.com/science/article/B6T84-3W2XGYY-5/2/d7ce60aa233f5402ceb7ca98cf7d6d43>

U.S. Census Bureau (2000) 'U.S. Interim Projections by Age, Sex, Race, and Hispanic Origin.' [online]. Available from <http://www.census.gov/ipc/www/usinterimproj/> [February 2009]

United Nations (2012) *Population Ageing and Development 2012 (Wall Chart).* New York, NY,: Population Division, Department of Economic and Social Affairs. Available from <http://books.google.co.uk/books?id=u8eBMAEACAAJ>

United Nations Population Division (2007) 'World Population Will Increase by 2.5 Billion by 2050; People over 60 to Increase by More Than 1 Billion.' [online]. Available from <http://www.un.org/News/Press/docs//2007/pop952.doc.htm> [February 2009]

Vaidyanathan, P. P. (2001) 'Generalizations of the Sampling Theorem: Seven Decades after Nyquist.' *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on* 48, (9) 1094-1109

Valdastri, P., Rossi, S., Menciassi, A., Lionetti, V., Bernini, F., Recchia, F. A. and Dario, P. (2008) 'An Implantable Zigbee Ready Telemetric Platform for in Vivo Monitoring of Physiological Parameters.' *Sensors and Actuators A: Physical* 142, (1) 369-378 available from <http://www.sciencedirect.com/science/article/B6THG-4NHV4FP-C/2/7da4124b62a734700fa25f37d224ee8b>

Van Boxtel, A. (2001) 'Optimal Signal Bandwidth for the Recording of Surface Emg Activity of Facial, Jaw, Oral, and Neck Muscles.' *Psychophysiology* 38, (1) 22-34

Van Halteren, A., Bults, R., Wac, K., Dokovsky, N., Koprinkov, G., Widya, I., Konstantas, D., Jones, V. and Herzog, R. (2004) 'Wireless Body Area Networks for Healthcare: The Mobihealth Project.' *Stud Health Technol Inform* 108, 181-193

Vapnik, V. N. (1995) *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc.

Vapnik, V. N. (1999) 'An Overview of Statistical Learning Theory.' *Neural Networks, IEEE Transactions on* 10, (5) 988-999

Venkatachalam, K. L., Herbrandson, J. E. and Asirvatham, S. J. (2011) 'Signals and Signal Processing for the Electrophysiologist: Part I: Electrogram Acquisition.' *Circulation: Arrhythmia and Electrophysiology* 4, (6) 965-973 available from <http://circep.ahajournals.org/content/4/6/965.short>

Vergados, D., Kavvadias, C., Bigalke, O., Eppler, A., Jerabek, B., Alevizos, A., Caragiozidis, M., Biniaris, C. and Robert, E. (2008) 'An Intelligent Interactive Healthcare Services Environment for Assisted Living at Home.' *Pervasive Computing Technologies for Healthcare, 2008. PervasiveHealth 2008. Second International Conference on*. Jan. 30 2008-Feb. 1 2008

Vert, J. P., Tsuda, K. and Schölkopf, B. (2004) 'A Primer on Kernel Methods.' *Kernel Methods in Computational Biology* 35-70

Von Lilienfeld-Toal, M., Midgley, K., Lieberbach, S., Barnard, L., Glasmacher, A., Gilleece, M. and Cook, G. (2007) 'Observation-Based Early Warning Scores to Detect Impending Critical Illness Predict in-Hospital and Overall Survival in Patients Undergoing Allogeneic Stem Cell Transplantation.' *Biology of blood and marrow transplantation : journal of the American Society for Blood and Marrow Transplantation* 13, (5) 568-576 available from <http://linkinghub.elsevier.com/retrieve/pii/S108387910700002X?showall=true>

Warr, P. (1990) 'The Measurement of Well-Being and Other Aspects of Mental Health.' *Journal of Occupational Psychology* 63, (3) 193-210 available from <http://search.ebscohost.com/login.aspx?direct=true&db=bth&AN=4617291&site=ehost-live>

Wei, J. and Kong, S. G. (2007) 'Block-Based Neural Networks for Personalized Ecg Signal Classification.' *Neural Networks, IEEE Transactions on* 18, (6) 1750-1761

Welch Allyn Inc (2009) *Micropaq® Wearable Monitor* [online] available from <http://www.welchallyn.com/products/en-us/x-11-ac-100-0000000001100.htm> [September 2009]

Williams, J. R. and Amaratunga, K. (1994) 'Introduction to Wavelets in Engineering.' *International Journal for Numerical Methods in Engineering* 37, (14) 2365-2388 available from <http://dx.doi.org/10.1002/nme.1620371403>

World Health Organisation (2010) *The World Health Report 2010: Health Systems Financing: The Path to Universal Coverage.* Geneva: World Health Organisation

World Health Organization (2003) *World Health Organization Definition of Health* [online] available from <http://www.who.int/about/definition/en/print.html> [July 2009]

Wu, K.-P. and Wang, S.-D. (2009) 'Choosing the Kernel Parameters for Support Vector Machines by the Inter-Cluster Distance in the Feature Space.' *Pattern Recognition* 42, (5) 710-717 available from <http://www.sciencedirect.com/science/article/pii/S0031320308003671>

Wu, W. H., Bui, A. a. T., Batalin, M. A., Au, L. K., Binney, J. D. and Kaiser, W. J. (2008) 'Medic: Medical Embedded Device for Individualized Care.' *Artificial Intelligence in Medicine* 42, 137—152

Xue, Q., Hu, Y. H. and Tompkins, W. J. (1992) 'Neural-Network-Based Adaptive Matched Filtering for Qrs Detection.' *Biomedical Engineering, IEEE Transactions on* 39, (4) 317-329

Yaghouby, F., Ayatollahi, A., Bahramali, R., Yaghouby, M. and Alavi, A. H. (2010) 'Towards Automatic Detection of Atrial Fibrillation: A Hybrid Computational Approach.' *Computers in Biology and Medicine* 40, (11-12) 919-930 available from <http://www.scopus.com/inward/record.url?eid=2-s2.0-78649329615&partnerID=40&md5=bc4157940634552b0f975ddf006c45b1>

Yeh, Y.-C. and Wang, W.-J. (2008) 'Qrs Complexes Detection for Ecg Signal: The Difference Operation Method.' *Comput. Methods Prog. Biomed.* 91, (3) 245-254

Yong, L. and Jiang Hong, Y. (2004) 'The Reduction of Noises in Ecg Signal Using a Frequency Response Masking Based Fir Filter.' *Biomedical Circuits and Systems, 2004 IEEE International Workshop on.* 1-3 Dec. 2004

Yuksel, M. and Dogac, A. (2011) 'Interoperability of Medical Device Information and the Clinical Applications: An Hl7 Rmim Based on the Iso/Ieee 11073 Dim.' *Information Technology in Biomedicine, IEEE Transactions on* 15, (4) 557-566

Zephyr Technology (2008) [online] available from <http://www.zephyr-technology.com/> [August 2009]

Zephyr Technology (2012) *Bioharness* [online] available from <http://www.zephyranywhere.com/products/bioharness-3/> [03.07.2013

Zhang, F. and Yong, L. (2009) 'Qrs Detection Based on Multiscale Mathematical Morphology for Wearable Ecg Devices in Body Area Networks.' *Biomedical Circuits and Systems, IEEE Transactions on* 3, (4) 220-228

Zhang, Y. (2007) 'Real-Time Development of Patient-Specific Alarm Algorithms for Critical Care.' *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*. 22-26 Aug. 2007

Zhang, Z., Dong, J., Luo, X., Choi, K.-S. and Wu, X. (2014) 'Heartbeat Classification Using Disease-Specific Feature Selection.' *Computers in Biology and Medicine* 46, (0) 79-89 available from <http://www.sciencedirect.com/science/article/pii/S001048251300348X>

Zidelmal, Z., Amirou, A., Adnane, M. and Belouchrani, A. (2012) 'Qrs Detection Based on Wavelet Coefficients.' *Computer Methods and Programs in Biomedicine* (0) available from <http://www.sciencedirect.com/science/article/pii/S016926071100321X>

Zigbee Alliance (n.d.) [online] available from <http://www.zigbee.org/> [February 2009]

Appendix 1: Ethical Approval

ETHICS REVIEW FEEDBACK FORM

(Review feedback should be completed within 10 working days)

Name of applicant: Jacek Lewandowski   .... **Faculty/School/Department:** EC
Research project title: Innovative use of mobile technology for well-being monitoring

Comments by the reviewer

---

1. Evaluation of the ethics of the proposal:

The proposal will use healthy adult human volunteers for a physiological modelling study using various body sensors. Informed consent will be applied. Monitoring methods already proposed in the literature, albeit recently, are being used together (as far as I can tell) with devices specifically designed for human physiological monitoring. The proposal needs a statement regarding the safety of the methods and devices. To best of expert knowledge at this time, do we know that this type of monitoring poses no or minimal health and safety risk? If not, what are the risks? A statement on this with back up reference would be useful. My other concern is that at this stage no time period for the wearing of the devices or indication of exact nature of the physical activity is given. It would be better to sate this or at least give upper limits.

Minor Conditions

- Data anonymity should be assured and a statement provided. I see such a statement in the participant information sheet but did not see it in the proposal.
- Use equipment for reputable sources specifically designed for health monitoring (as these will have already had safety checks) and add statement on this.
- Follow instructions of usage for the devices selected and add statement on this in the proposal
- Provide a statement on health and safety risks
- Give likely upper limits of device wearing times and physical activities or at least a statement that these will be within the bounds of the recommended usage of the devices.

---

2. Evaluation of the participant information sheet and consent form:

These are okay provided the timing for wearing devices and type of required physical activity is added before these are put into use and the above conditions can be met.

---

3. Recommendation:
(Please indicate as appropriate and advise on any conditions. If there any conditions, the applicant will be required to resubmit his/her application and this will be sent to the same reviewer).

☐	Approved - no conditions attached
☐	Approved with minor conditions (no need to resubmit)
☐	Conditional upon the following – please use additional sheets if necessary (please re-submit application)
☐	Rejected for the following reason(s) – please use other side if necessary
☐	Further advice/notes - please use other side if necessary

Name of reviewer: Anne James
Signature: ....
Date: 4ᵗʰ December 2009

Appendix 2: Publications

A3.1 Analog-to-Digital Converter Interface configuration and usage

I/O PINs and configuration

In order to allow the ADC conversion on particular sensor, the respective ADC port pins must be configured. It is done by selecting the peripheral module function and switching to input direction on the selected port pin. This process performs automatically before the conversion starts, and immediately after the conversion finish, it switches back to I/O function mode. The configuration is based on the information a developer provides, by implementing `msp430adc12 _channel_config_t struct` data type that holds all information needed to configure the ADC12 for single channel conversions. The parameters that have to be specified, following MSP430 User Guide (Texas Instruments 2006), present as follows:

- `.inch`: ADC12 input channel. An external input channel maps to one of msp430's A0-A7 device specific pins;

- `.sref`: reference voltage. If `REFERENCE_VREFplus_AVss` or `REFERENCE_VREFplus_VREFnegterm` is chosen and the client connects to the `Msp430Adc12ClientAutoRVGC` or `Msp430Adc12ClientAutoDMA_RVGC` component then the reference voltage generator is automatically enabled to the voltage level defined by the `"ref2_5v"` flag (see below) when the `Resource.granted()` event is signalled to the client. Otherwise this flag is ignored;

- `.ref2_5v`: Reference voltage level generator. (See `"sref"` above);

- `.adc12ssel`: ADC12 clock source selected for the sample-hold-time clock. The combination of `"adc12ssel"`, `"adc12div"` and `"sht"` defines the final sample-hold-time. The sample-hold-time depends on the resistance of the attached sensor and is calculated using to the formula discussed in section 17.2.5 Sample and Conversion Timing of the MSP430 User Guide (Texas Instruments 2006);

- `.adc12div`: ADC12 clock divider. (See usage in `"adc12ssel"` above);

- `.sht`: Sample-hold-time defines the hold time, expressed in jiffies. (See usage in `"adc12ssel"` above);

- `.sampcon_ssel`: Clock source for the sampling period. When an ADC client specifies a non-zero "jiffies" parameter, the ADC implementation will automatically configure timer to be sourced from `"sampcon_ssel"` with an input divider of `"sampcon_id"`. During the sampling process timer will be used to trigger a single `Msp430Adc12SingleChannel` interface or a sequence of `Msp430Adc12MultiChannel` interface conversions every "jiffies" clock ticks.

- .sampcon_id: Input divider for "sampcon_ssel". (See "sampcon_ssel" above)

The sample, full type definition of `msp430adc12_channel_config_t` struct presents as follows:

```c
typedef struct {
  unsigned int inch: 4;          // input channel
  unsigned int sref: 3;          // reference voltage
  unsigned int ref2_5v: 1;       // reference voltage level
  unsigned int adc12ssel: 2;     // clock source sample-hold-time
  unsigned int adc12div: 3;      // clock divider sample-hold-time
  unsigned int sht: 4;           // sample-hold-time
  unsigned int sampcon_ssel: 2;  // clock source sampcon signal
  unsigned int sampcon_id: 2;    // clock divider sampcon signal
} msp430adc12_channel_config_t;
```

Sampling Process

Sampling a channel is performed by calling a sequence of two commands, `configureX()` and `getData()`, where X is either `'Single'`, `'SingleRepeat'`, `'Multiple'` or `'MultipleRepeat'`. Conversion results are signalled by the `dataReadySingle()` or `dataReadyMultiple()` event, depending on the previous configuration. There are four possible sequences of configuration which are followed by different events signalled back to the client by the ADC subsystem. These are:

- `configureSingle()` & `getData()` -> `singleDataReady()`,
- `configureSingleRepeat()` & `getData()` -> `singleDataReady()`,
- `configureMultiple()` & `getData()` -> `multipleDataReady()`,
- `configureMultipleRepeat()` & `getData()` -> `multipleDataReady()`,

Such configurations are valid until the client reconfigures or releases the ADC resource, except for `configureMultipleRepeat()`, which is only valid for a single call to `getData()`. This means that after a successful configuration with, for example, `configureSingle()` the client may call `getData()` more than once without reconfiguring the ADC. For more information on operation of MSP430 Analog-to-Digital converter from HAL and HIL perspective refer to TEP 101 specification (Hauer, Levis, Handziski *et al.* 2006).

A3.2 Analog-to-Digital Converter Interface configuration and usage

Name, Return Type, and Arguments	Description
`public abstract NodeConnection getConnection();`	Returns `NodeConnection` object associated with this `Node` that implements node specific signalling protocol.
`public abstract Frame getFrame(boolean accessCode);`	If the `accessCode` is `true` fetch the `Frame` object dynamically through direct node negotiation (if the node supports such) otherwise return statically the default node frame object associated with this `Node` object. *) Implementation of a Node specific negotiation mechanism at this point.*
`public abstract NodeInfo getNodeInfo(boolean accessCode);`	Returns `NodeInfo` that contains information about the sensor node. If `accessCode` is `true` than performs a remote retrieval of data from the physical node otherwise returns pre-known info only. *) Implementation of a `CommStack` specific remote node information retrieval required at this point.*
`public abstract SensorInfo[] getSensors(boolean accessCode);`	Returns an array of `SensorInfo` objects representing available sensors. This method search for sensors with the specified inquiry access code. If `accessCode` is `true` than performs a remote search at the physical node level, otherwise returns pre-known sensors only. 0..* objects can exist. If `SensorInfo[]` is `null` than node does not provide data acquisition service and is used as a network repeater. *) Implementation of a Node specific command required at this point to perform the search.*
`public boolean isAvailable();`	Returns `true` when sensor node is available else returns `false`. *) Implementation of a Node specific pinging mechanism required at this point.*

Table A3.1 The abstract methods to implement the `org.j2me4wsn.network.Node` interface.

Name, Return Type, and Arguments	Description
`public BluetoothNodeInfo(<parameters>)`	Constructor of `BluetoothNodeInfo` object. Because `NodeInfo` is an abstract class it can't have an instance. Implementation of node specific `NodeInfo` object must provide at least one object constructor.
`public abstract Node getNode();`	Returns the stack specific `Node` object based on information stored in `NodeInfo` object about physical sensor node. The Node object is used to establish the connection using correct node communication stack.

Table A1.2 The abstract methods to implement the `org.j2me4wsn.network.NodeInfo`.

Name, Return Type, and Arguments	Description
`public BluetoothNodeConnection()`	Constructor of `BluetoothNodeConenction` object. Because `NodeConnection` is an abstract class it can't have an instance. Implementation of node specific `NodeConnection` object must provide at least one object constructor.
`public abstract void open(NodeInfo nodeinfo)`	Creates and opens a connection using `Conenctor.open(url)` using `NodeInfo` connection information. It opens input and output data streams too.
`public abstract void write(Frame qry)`	Writes `Frame` object to the node output stream. The general contract for `write(Frame)` is that it should send `Frame` object to the node and wait for confirmation of success.
`public abstract Frame read()`	Reads `Frame` object from the input stream. The general contract for `read()` is that it should read full `Frame` object send by the node and return it .
`public abstract void close()`	Closes the connection and associated data streams.

Table A3.3The abstract methods to implement the `org.j2me4wsn.network.NodeConnection`.

A3.3 Compatibility issues between J2SE and CLDC platforms

J2SE platform	CLDC platform
java.lang.*;	
Cloneable	N/A *) The `Cloneable` interface is removed. For each class to clone a custom `clone()` method is implemented.
Comperable	org.j2megloss.Comperable; *) In order to enable objects comparison the following `Comperable` interface was defined. Each class that will be compared must implement this interface: ```\npublic interface Comperable {\n public int compareTo(Object o);\n}\n```

Math
  − **exp**(double a)
  − **log**(double a)
  − **random**()
  − **round**(float a)
	org.j2megloss.MathUtil
	  + **exp**(double a)
	  + **log**(double a)
	  + **random**()
	  + **round**(float a)
	
	*) The `MathUtil` class implements static methods which provide missing math functions. The methods base on the J2SE `java.lang.Math` methods with minor amendments.

java.util.*;	
Collection List ArrayList TreeSet	java.util.Vector *) `Collections` are replaced by the `Vector` container class. All references to the old `Collection` objects are modified to follow `Vector` interface methods.
EventObject	org.j2megloss.EventObject *) The custom EventObject is implemented that carries the data object and a reference to a source object on which the event initially occurred. The class base on its J2SE version.
EventListener	N/A *) The `EventListener` is removed. It is only a tagging interface that all event listener interfaces extends.
Iterator	java.util.Enumeration

StringTokenizer	`org.j2megloss.StringTokenizer` *) The `StringTokenizer` class breaks a string into tokens. The class base on its J2SE version.
java.io.*;	
`FileInputStream` `FileOutputStream`	`java.io.InputStream` `java.io.OutputStream`

`File`
`FileNotFoundException`
	`javax.microedition.io.Connector`
	`javax.microedition.io.file.FileConnection`
	
	*) In order to use this class the device must implement JSR 75 optional package. `FileConnection` is a simple, lightweight and very useful set of file-system APIs for the CLDC platform. It enables J2ME-based applications to create, read, and write files and directories located on mobile devices and external memory cards. The following code opens a `InputStream` from a file:
	
	```FileConnection inputFile =```
	``` (FileConnection) Connector.open(fileName);```
	```InputStream fis =```
	```inputFile.openInputStream();```

`FileReader`	`org.j2megloss.FileReader` *) Simple lightweight class that extends the `InputStreamReader`. `FileReader` is meant for reading streams of characters. The class base on its J2SE version.
`BufferedReader`	`import org.j2megloss.BufferedReader` *) Reads text from a character-input stream and buffer characters. It provides a readLine() from file method. The class base on its J2SE version.
`LineNumberReader`	`org.j2megloss.LineNumberReader` *) A buffered character-input stream that keeps track of line numbers used by the `InputStreamTokenizer` class. The class base on its J2SE version.
java.net.*;	

`URL`	`java.util.String`
	`javax.microedition.io.Connector`
	`javax.microedition.io.HttpConnection`
	`javax.microedition.io.file.FileConnection`
	
	*) Depending on the use of URL wrapper it is replaced by:
	• `String` if and URL serves as a path container only;
	• `String`, `Connector`, `HttpConnection` or `FileConnection` if the URL is responsible for

	establishing the connection with external resource such as http or file.
`java.sql.*;`	`org.j2megloss.db` *) CLDC platform does not support JDBC APIS. To connect to databases from mobile devices a remote web server is required. The `org.j2megloss.db` provides a lightweight interface to connect to a JAVA EE Servlet where queries and other database functionalities need to be implemented. The db package comes with a sample Servlet application that can be deployed on any Java server such as Apache Tomcat, and connected to database such as MySql and others.
java.awt.*;	
`Image;` `image.BufferedImage;`	`javax.microedition.lcdui.Image` *) The `Image` class is used to hold graphical image data. Image objects exist independently of the display device. They exist only in off-screen memory and will not be painted on the display unless an explicit command is issued by the application. All CLDC implementations are required to support images stored in the PNG format. In order to create an Image it use one of the available `createImage()` methods.

`image.PixelGrabber`
  – **grabPixels**()
	`javax.microedition.lcdui.Image`
	
	  + **getRGB**(int[] rgbData, int offset,
	                     int scanlength, int x,
	                     int y, int width,int height)
	
	*) The `getRGB()` method, similar as `grabPixels()` method, obtains ARGB pixel data from the specified region of this image and stores it in the provided array of integers. Each pixel value is stored in 0xAARRGGBB format, where the high-order byte contains the alpha channel and the remaining bytes contain colour components for red, green and blue, respectively. The alpha channel specifies the opacity of the pixel, where a value of 0x00 represents a pixel that is fully transparent and a value of 0xFF represents a fully opaque pixel. The `Image` class belongs to a MIDP 2.1 profile.

javax.imageio.*;	
`ImageIO.read(theFile)`	`javax.microedition.lcdui.Image;` `*)` In order to create an `Image` from a file a `createImage(InputStream stream)` method can be used, passing a `InputStream` from a File as a parameter.
`ImageIO.write()`	`org.j2megloss.PngEncoder;` `*)` `PngEncoder` takes a Java Image object and creates a byte string which can be saved as a PNG file. The class use `jazzlib`, a `java.util.zip` support for CLDC platform