Automatic classification of colonic histophathological images using adaptive neuro-fuzzy networks and genetic algorithms

Gan Lim , Laurence A.

Award date:
2011

Awarding institution:
Coventry University

COVENTRY UNIVERSITY

AUTOMATIC CLASSIFICATION OF COLONIC HISTOPHATHOLOGICAL IMAGES USING ADAPTIVE NEURO-FUZZY NETWORKS AND GENETIC ALGORITHMS

Laurence A. Gan Lim

Doctor of Philosophy

October 2011

DEDICATION

To my parents,

Leoncio

and

Corazon

ABSTRACT

This research is focused on the design, development, implementation, and evaluation of a hybrid classifier system that discriminates between three (3) classes of colonic histopathological images namely, normal, adenomatous polyp, and cancerous lesions. Here, a hybrid classifier system is realised by combining and using fuzzy logic, artificial neural networks and genetic algorithms to tackle the classification problem. The implementation of the solution to the problem has been divided into two parts: feature selection and classification. The scope of the study is focused on the use of textural features introduced by Haralick, as input to the classifier system. Variance ratios derived from scatter matrices and genetic algorithms are the tools used and compared in order to select candidate feature sets. A Kohonen self-organising map is used in the fitness function of the genetic algorithm. Results show that the use of variance ratio derived from scatter matrices is far simpler and faster than the use of a genetic algorithm with the Kohonen map. In the classification part of this study, a hybrid neuro-fuzzy adaptive network, known as Adaptive Network-Based Fuzzy Inference System, or ANFIS, is used. The elegance and power of this computational framework is clearly evident as the different network parameters and fuzzy membership functions are adaptively adjusted, given simply the data from the feature sets. It is later pointed out in this thesis that the confusion matrix is an effective presentation format of the performance of a classifier but lacks certain important details regarding the shortcomings of a particular classifier that is being evaluated. This study proposes the use of a Mean Relative Difference Confusion Matrix, or MRDCM, a name coined in this study. MRDCM can be thought of as a modified version of the conventional confusion matrix. Instead of counting the number of correct classifications and misclassifications, MRDCM tabulates the average differences between expected and predicted real

number output values of the Sugeno-type defuzzification of the ANFIS. Another performance indicator that is introduced in this research is a parameter which is coined to be known as Classification Performance Index, or CPI. The advantage of using CPI is that it is simply a single number similar to accuracy percentage, a value that one would normally obtain when the sum of the leading diagonal of a confusion matrix is calculated and normalised. Although the CPI is slightly more complicated to compute, it definitely accounts for the misclassifications produced by a classifier under scrutiny. The CPI is calculated by multiplying each cell of the confusion matrix by performance factors that either increase or decrease a particular number, depending on its location in the confusion matrix. It is believed that performance indicators of classifiers are as important and as crucial as the classifier algorithms themselves since these parameters allow us to truly measure the success and failure of our solutions.

ACKNOWLEDGEMENT

First and foremost, I would like to express my utmost gratitude to my supervisor, Prof. Raouf Naguib, for investing his time and talent in making certain that I succeed in my doctorate studies. I will not forget his numerous visits to Manila to monitor my progress and assist me in solving some of my problems in research. I am also indebted to Prof. Ian Marshall for making it possible for me to get a special financial arrangement with the University regarding my tuition and fees.

My external sponsors, Dr. Leornardo A. Gan Lim and Dr. Leonil A. Gan Lim, both made contributions to my scholarship fund and other fees related to my studies. I will always remember to do very well in everything that I do in life in order to make every dollar or pound they spent for me worth it.

I should also thank Prof. Alvin Culaba for introducing me to my supervisor and for his invaluable advice about my studies and professional career. Equally deserving of similar gratitude is Prof. Elmer Dadios who acted as my local supervisor in Manila. His insights and ideas inspired me to design my algorithms. He was also the one who introduced me to the area of Soft Computing. I am proud to have been one of his students.

This research would not have been possible if Dr. Jose Marie Avila and Dr. Debbie dela Fuente had not provided me with the colonic microscopic images that served as the input data set to my algorithms. To them, I say thank you from the bottom of my heart. You truly are heaven-sent.

I would also like to express gratitude to Prof. Julius Maridable, Prof. Pag-asa Gaspillo, and Prof. Archie Maglaya for their support and encouragement while I did my research work at De La Salle University in Manila. The same goes for all my colleagues at the Mechanical Engineering Dept. of De La Salle University.

Last but not least, I would like to thank the rest of my family, especially my mother Corazon, my partner Lorelei, and my daughter Sophia for their patience and love while I dedicated countless number of hours away from home to focus on my research.

TABLE OF CONTENTS

DEDICATION i

ABSTRACT ii

ACKNOWLEDGEMENT iv

LIST OF FIGURES ix

LIST OF TABLES xv

GLOSSARY xviii

Chapter 1 - INTRODUCTION 1

1.1 Background Literature and Origins of the Research 1
1.2 Statement of the Problem 8
1.3 The Proposed Solution 10
1.4 Aims of the Study 13
1.5 Novel Contributions to the State of the Art 14
1.6 Structure of the Thesis 18

Chapter 2 - WHAT IS COLON CANCER? 20

Chapter 3 – REVIEW OF IMAGE ANALYSIS AND ALGORITHMS USED 34

3.1 Digital Image 34
3.2 Histogram Equalisation 36
3.3 Unsharp Masking 40
3.4 Haralick Textural Features 42
3.5 Scatter Matrices and Boland *et al.* (1998) Variance Ratio 47
3.6 Kohonen Self-Organising Map (KSOM) 51
3.7 Genetic Algorithms (GA) 52
3.8 Adaptive-Network-Based Fuzzy Inference System (ANFIS) 56
3.9 Confusion Matrix 60
3.10 Software and Hardware Used 61

Chapter 4 – TEXTURAL FEATURE CALCULATION AND FEATURE SELECTION 62

4.1 Production of Digital Images from Microscopic Slides 62

4.2 Feature Selection Using Variance Ratio 65
4.3 Feature Selection Using Genetic Algorithm and Kohonen Self-Organising Map 73
4.4 Preparing for Image Classification 82
Chapter 5 – AUTOMATIC CLASSIFICATION OF IMAGES 84
5.1 Mean Relative Difference Confusion Matrix (MRDCM) and Classification Performance Index (CPI) 84
5.2 Implementation of ANFIS 90
Set A feature combination [Mean, Sum average, and Sum variance]: 92
Set B feature combination [Mean and Sum average]: 96
Set C feature combination [Contrast, entropy, and difference variance]: 100
Set D feature combination [Contrast, IDM, and sum variance]: 104
Set E feature combination [Sum average and difference entropy]: 108
Set F feature combination [Contrast, inverse difference moment or IDM, and difference variance]: 113
5.3 Image Classification by Human Pathologists 117
5.4 Summary of the Image Classification Implementation 120
5.5 Comparison of Results with Previous Studies 123
Chapter 6 - CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK 125
REFERENCES 130
APPENDICES 138
A.1 TRAINING AND TEST *.data: 138
“trainingData1200x1600.data” – training data for image size of 1200x1600 pixels, 16 quantisation levels: 138
A.2 TRAINING AND TEST *.dat FILES 144
“trainDataANFIS_mean_sumAve_sumVar.dat” – DAT file containing the properties mean, sum average, and sum variance: 144
A.3 MATLAB program “image2FeatureDATAFile.m” used to calculate the textural properties of each training image and store in a .data file: 148
A.4 MATLAB function “glf_computeVarianceRatio.m” used to calculate the variance ratio of each textural property for the entire training image set: 151
A.5 MATLAB function “glf_SOMFitnessFunction.m” as the fitness function used in the MATLAB GA Toolbox 153
A.6 MATLAB program “writeToFileChosenPropertiesForANFIS.m” to generate the training and test *.dat files from the *.data files. 155

A.7 MATLAB program “ANFISthesisImplementationCommandLine.m” to implement ANFIS and produce the necessary classification results..... 158

A.8 Image Classification Test for Pathologists:..... 163

A.9 Expected classes of the images used in the classification test for pathologists: 171

A.10 Summary of Publications Produced Based on this Research 172

LIST OF FIGURES

Figure 2.1 Structural differences between normal and cancerous cells 21

Figure 2.2 Mutation of proto-oncogene into oncogene 22

Figure 2.3 Different types of cancer 23

Figure 2.4 Normal and Cancer Cell Division 24

Figure 2.5 New bowel cancer cases and age-specific incidence rates by sex in the UK for 2005 26

Figure 2.6 Cross-section of the Colon (National Cancer Institute, 2008f) 28

Figure 2.7 T1 tumour invasion of the colonic tissue (Greene *et al.* 2006: 111) 28

Figure 2.8 T2 tumour invasion of the colonic tissue (Greene *et al.* 2006: 111) 29

Figure 2.9 T3 tumour invasion of the colonic tissue (Greene *et al.* 2006: 112) 29

Figure 2.10 T4 tumour invasion of the colonic tissue (Greene *et al.* 2006: 113) 30

Figure 2.11 Appearance of polyps in the colon (National Cancer Institute, 2008h) 32

Figure 2.12 Colorectal Cancer Staging (National Cancer Institute, 2008i) 33

Figure 3.1 Coordinate convention for a digital image 35

Figure 3.2 Histogram of the given image in the example before histogram equalisation. 38

Figure 3.3 Histogram of the given image in the example after histogram equalisation. 40

Figure 3.4 Schema of unsharp masking using Laplacian mask 41

Figure 3.5 Relationship between the Laplacian mask and the unsharp mask. α is the 41

Figure 3.6 Equivalent plot of the variance ratios in Table 3.3 for the given illustrative example. 50

Figure 3.7 Schematic representation of the Kohonen self-organising network or KSON 51

Figure 3.8 Schematic flowchart of a genetic algorithm 54

Figure 3.9 (a) and (b) 2-input 1st order Sugeno fuzzy model with 2 rules and the equivalent ANFIS architecture based on Fig. 28 of Jang and Sun (1995) 57

Figure 4.1 Schematic diagram of the imaging system 63

Figure 4.2 Variance ratio bar graph for training images with 300x400 pixels image size and 32 quantisation levels 68

Figure 4.3 Variance ratio bar graph for training images with 300x400 pixels image size and 24 quantisation levels 68

Figure 4.4 Variance ratio bar graph for training images with 300x400 pixels image size and 16 quantisation levels 69

Figure 4.5 Variance ratio bar graph for training images with 300x400 pixels image size and 8 quantisation levels 69

Figure 4.6 Variance ratio bar graph for training images with 1200x1600 pixels image size and 16 quantisation levels 70

Figure 4.7 Variance ratio bar graph for training images with 900x1200 pixels image size and 16 quantisation levels 70

Figure 4.8 Variance ratio bar graph for training images with 600x800 pixels image size and 16 quantisation levels 71

Figure 4.9 Variance ratio bar graph for training images with 300x400 pixels image size, 16 quantisation levels, and histogram equalised 71

Figure 4.10 Variance ratio bar graph for training images with 300x400 pixels image size, 16 quantisation levels, and unsharp masking coefficient α = 0.5 72

Figure 4.11 Schematic diagram of the GA-KSOM feature selector 74

Figure 4.12 Penalty function within the GA fitness function 75

Figure 4.13 Fitness values for run #1 of the GA-KSOM search algorithm..... 77

Figure 4.14 Feature coefficients from run #1 of the GA-KSOM search algorithm 78

Figure 4.15 Fitness values for run #2 of the GA-KSOM search algorithm..... 78

Figure 4.16 Feature coefficients from run #2 of the GA-KSOM search algorithm 79

Figure 4.17 Fitness values for run #3 of the GA-KSOM search algorithm..... 79

Figure 4.18 Feature coefficients from run #3 of the GA-KSOM search algorithm 80

Figure 4.19 Fitness values for run #4 of the GA-KSOM search algorithm..... 80

Figure 4.20 Feature coefficients from run #4 of the GA-KSOM search algorithm 81

Figure 5.1 ANFIS Structure of Set A features..... 92

Figure 5.2 ANFIS Membership Functions using Set A features: Mean (input1), Sum average (input2), Sum variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'..... 93

Figure 5.3 ANFIS root mean squared errors during training for Set A features: Mean, Sum average, Sum variance 94

Figure 5.4 Classification performance trained ANFIS using training and testing data sets for Set A features: Mean, Sum average, Sum variance..... 94

Figure 5.5 Classification Difference of Trained ANFIS using training and testing data sets for Set A features: Mean, Sum average, Sum variance..... 95

Figure 5.6 ANFIS Structure of Set B features..... 96

Figure 5.7 ANFIS Membership Functions using Set B features: Mean (input1), Sum average (input2). Left side plots are refer to 'before training' while the right side plots refer to 'after training'..... 97

Figure 5.8 ANFIS root mean squared errors during training for Set B features: Mean and Sum average 97

Figure 5.9 Classification performance trained ANFIS using training and testing data sets for Set B features: Mean and Sum average 98

Figure 5.10 Classification Difference of Trained ANFIS using training and testing data sets for Set B features: Mean and Sum average 98

Figure 5.11 ANFIS Structure of Set C features..... 100

Figure 5.12 ANFIS Membership Functions using Set C features: contrast (input1), entropy (input2), difference variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'..... 101

Figure 5.13 ANFIS root mean squared errors during training for Set C features: contrast, entropy, difference variance..... 102

Figure 5.14 Classification performance trained ANFIS using training and testing data sets for Set C features: contrast, entropy, difference variance..... 102

Figure 5.15 Classification Difference of Trained ANFIS using training and testing data sets for Set C features: contrast, entropy, difference variance 103

Figure 5.16 ANFIS Structure of Set D features..... 104

Figure 5.17 ANFIS Membership Functions using Set D features: contrast (input1), inverse difference moment or IDM (input2), sum variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'. 105

Figure 5.18 ANFIS root mean squared errors during training for Set D features: contrast, inverse difference moment or IDM, sum variance 106

Figure 5.19 Classification performance trained ANFIS using training and testing data sets for Set D features: contrast, inverse difference moment or IDM, sum variance 106

Figure 5.20 Classification Difference of Trained ANFIS using training and testing data sets for Set D features: contrast, inverse difference moment or IDM, sum variance 107

Figure 5.21 ANFIS Structure of Set E features..... 108

Figure 5.22 ANFIS Membership Functions using Set E features: sum average (input1), difference entropy (input2). Left side plots are refer to 'before training' while the right side plots refer to 'after training'..... 109

Figure 5.23 ANFIS root mean squared errors during training for Set E features: sum average, difference entropy 110

Figure 5.24 Classification performance trained ANFIS using training and testing data sets for E features: sum average, difference entropy..... 111

Figure 5.25 Classification Difference of Trained ANFIS using training and testing data sets for Set E features: sum average, difference entropy 111

Figure 5.26 ANFIS Structure of Set F features 113

Figure 5.27 ANFIS Membership Functions using Set F features: contrast (input1), inverse difference moment or IDM (input2), difference variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'. 114

Figure 5.28 ANFIS root mean squared errors during training for Set F features: contrast, inverse difference moment or IDM, difference variance 115

Figure 5.29 Classification performance trained ANFIS using training and testing data sets for Set F features: contrast, inverse difference moment or IDM, difference variance..... 115

Figure 5.30 Classification Difference of Trained ANFIS using training and testing data sets for Set F features: contrast, inverse difference moment or IDM, difference variance 116

Figure 5.31 Comparison between the classification accuracy and classification performance index (CPI) of the pathologists and ANFIS algorithms using texture properties. 119

Figure 5.32 Average accuracy and CPI values for the pathologists and the ANFIS algorithm with different feature sets..... 120

Figure 5.33 Comparison of classification performances of the different feature combinations used in this research using the test image set and ANFIS as classifier. The CPI is the classification performance index while the accuracy is the decimal version of the percent accuracy. 122

LIST OF TABLES

Table 2.1 Summary of the TNM staging system.....27

Table 2.2 AJCC Stage Groupings.....27

Table 2.3 Duke's Staging System for Colorectal Cancer.....30

Table 2.4 Survival rates for colon cancer by stage (American Cancer Society, 2008).....30

Table 2.5 AJCC cancer grading system.....31

Table 3.1 Calculation of the new histogram of the image in the given illustrative example on histogram equalisation. 38

Table 3.2 Illustrative example on how to calculate the variance ratios used by Boland et al. (1998).....49

Table 3.3 The calculated variance ratios in the give illustrative example in Table 3.2.....50

Table 3.4 Activities in each pass in the ANFIS hybrid learning procedure.....58

Table 3.5 Example of a confusion matrix with 3 classes. The columns are the expected classifications while the rows are the predicted classes of the classifier. 60

Table 4.1 Variance ratios of the training images with different sizes, ...67

Table 4.2 Parameters and settings used in the implementation of GA 76

Table 4.3 Table of values of feature coefficients obtained from GA-KSOM search algorithm with elapsed times and best (minimum) fitness values 76

Table 5.1 General format of an MRDCM or Mean Relative Difference Confusion Matrix. The elements a, e, and i are the main diagonal elements. The rest of the elements are the off-diagonal elements. 86

Table 5.2 Format of the confusion matrix used in this study. 88

Table 5.3 Format of the factor matrix. The letters assigned to each element of the matrix correspond to the left column of Table 5.4 and to the entries in Table 5.2 as multipliers. 88

Table 5.4 The suggested ranking of the elements of the factor matrix with the multiplying factors. Match the letters on the left column to the entries in Table 5.3. 88

Table 5.5 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set A features: Mean, Sum average, Sum variance 95

Table 5.6 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set A features: Mean, Sum average, Sum variance with threshold values of 0.25 and 0.75 95

Table 5.7 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set B features: Mean and Sum average 99

Table 5.8 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set B features: Mean and Sum average with threshold values of 0.25 and 0.75 99

Table 5.9 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set C features: contrast, entropy, difference variance 103

Table 5.10 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set C features: contrast, entropy, difference variance with threshold values of 0.25 and 0.75 104

Table 5.11 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set D features: contrast, inverse difference moment or IDM, sum variance 107

Table 5.12 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set D features: contrast, inverse difference moment or IDM, sum variance 108

Table 5.13 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set E features: sum average, difference entropy 112

Table 5.14 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set E features: sum average, difference entropy 112

Table 5.15 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set F features: contrast, inverse difference moment or IDM, difference variance 116

Table 5.16 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set F features: contrast, inverse difference moment or IDM, difference variance with threshold values of 0.25 and 0.75 117

Table 5.17 Results of the test survey conducted on 6 pathologists using 15 monochromatic colonic images selected randomly from the test image set. CPI stands for classification performance index. 118

Table 5.18 Summary of the normalized confusion matrices of the different feature combinations using the test image set and ANFIS as classifier. The columns represent the expected classifications while the rows are the predicted classifications. From left to right and from top to bottom, the classes are normal, adenomatous polyp, and cancerous. 122

GLOSSARY

Adenocarcinoma – a malignant tumour on secretory epithelium

Adenomatous polyp – benign tumour growth on the mucous surface considered to be precursor to cancer

AJCC – American Joint Committee on Cancer

AI – Artificial Intelligence

ANFIS – Adaptive Neuro-Fuzzy Inference System or Adaptive-Network-Based Fuzzy Inference System

ANN – Artificial Neural Network, or simply Neural Network (NN)

Antioncogene – genes also known as tumour suppressor genes that counter the effects of excessive cell proliferation

AI – Artificial Intelligence

BPNN – Back-Propagation Neural Network

Cancerous – malignant and invasive growth or tumour in tissue

Carcinoma – probably the most common type of cancer; cancers under this type arise from the cells that cover external and internal body surfaces

CIS – carcinoma in situ

Colon – large intestine

Colon cancer – cancer that starts in the large intestines or colon; sometimes called bowel cancer

Colonic – pertaining to the colon

CPI – Classification Performance Index

Cytoplasm – cell substance between the cell membrane and the nucleus

Dysplastic – abnormal tissue

FL – Fuzzy Logic

GLCM – grey-level co-occurrence matrix

GA – Genetic Algorithm/s

Histopathologic – pertaining to the microscopic examination of tissue

ICA – Independent Component Analysis

KSOM – Kohonen Self-Organising Map

LDA – Linear Discriminant Analysis

Leukemia – cancer that involves the blood cells and the bone marrow

LVQ – Learning Vector Quantization

Lymphoma – cancer that arises in the lymph nodes and tissues of the body's immune system

MRDCM – Mean Relative Difference Confusion Matrix

NCI – National Cancer Institute

NHS – National Health Service

NN – see ANN

Oncogenes – genes that promote hyperactive cell growth and division, resistance to cell death, and invasion of other parts of the tissue or body

PCA – Principal Component Analysis

Polyp – a growth in a mucous surface

PNN – Probabilistic Neural Network

Proto-oncogenes – genes which are responsible for the regulation of cell growth and differentiation

Sarcoma – cancer that arises from cells belonging to the supporting tissues of the body e.g., bone, cartilage, fat, connective tissue and muscle

SVM – Support Vector Machine

UICC – International Union Against Cancer

WHO – World Health Organization

Chapter 1 - INTRODUCTION

Cancer ranks third among the leading causes of morbidity and mortality in the Philippines (Ngelangel and Wang, 2002). Colon cancer, in particular, is among the leading types of cancer. Worldwide, colorectal cancer is considered the third most common neoplasm (Shuttleworth *et al.*, 2005). Similar to other types of cancer, early detection of cancer of the colon is key to a successful treatment. Traditionally, pathologists use a microscope to examine histopathological images of biopsy samples taken from patients and make judgments based on their professional expertise. Typically, a pathologist would make observations on some key features in the image and subsequently be able to classify whether or not the tissue under examination contains abnormality. Since this procedure is performed by a human expert, it is therefore subject to inconsistencies due to factors that might affect human performance. To overcome this problem, it has been proposed to use mathematical and artificial intelligence (AI) paradigms to aid in the analysis of medical images, such as histopathological images of colon tissues.

1.1 Background Literature and Origins of the Research

Considerable research has been undertaken over the past two decades in an effort to automate cancer diagnosis (Demir and Yener, 2005). Some of the early implementations of colonic microscopic image classifiers using computers were

developed by Hamilton *et al*. in 1987. In this study, a semi-automatic image analysis was implemented with a morphological assessment of 15 nuclear and cellular variables in normal (n=20) and malignant (n=30) epithelium. Principal Component Analysis was employed to identify the four main sources of variation within the dataset: nuclear size, nuclear cytoplasmic ratio and nuclear position within the cell; the variability of nuclear size; nuclear elongation and polarity; and nuclear shape and its variation. Discriminant Analysis was performed using normal mucosa and adenocarcinoma in ulcerative colitis as classifications or groupings with 10 normal mucosa samples and 20 adenocarcinoma samples. The authors claimed a perfect discrimination of the samples from the two groups. The mean nuclear cytoplasmic ratio and the coefficient of variation of nucleus to cell apex distance were chosen as discriminating features through stepwise variable selection.

Research in the classification of microscopic images of colonic mucosa has shown that textural features derived from a grey-level co-occurrence matrix or GLCM are very useful. Esgiar *et al.* (1999) analyzed 44 normal and 58 cancer images captured to a computer via microscope with a CCD camera. Entropy and correlation were the two types of texture features that were used in this study. To improve classification, fractal analysis was incorporated. It was reported that a classification accuracy of 94% was achieved. The classification methods used in the study were linear discriminant function and k-nearest neighbour (k=2). In 2001, Atlamazoglou *et al.* used GLCMs to extract features from a total of 70 fluorescence microscope images of colonic tissue sections stained with a novel selective fluoroprobe. Directional GLCMs for each image were combined into a non-dimensional GLCM by averaging values from four angular directions [0^o, 45^o, 90^o, 135^o] with a distance of 1 pixel. From nine textural features, four features were selected and used to describe and classify each fluorescence image: inverse difference moment, correlation, and the two information measures of correlation: f_{12} and f_{13} in Haralick *et al.* (1973). The selection of features was based on

a modified version of the multiple discriminant analysis criterion. The chosen four features were selected since they had the highest mean and standard deviation values in the analysis. To distinguish between healthy and adenocarcinomatous colonic mucosa, the authors made use of a Mahalanobis distance linear discriminant classifier and a method based on a 'score' of an image proposed in 1996. As a result, 95% of the images were correctly classified. Shuttleworth *et al.* (2002a) proposed to use colour texture analysis in classifying colon cancer images. The study reported that classification using colour texture offered an improvement over classification based solely on grey-level texture. The authors explained that the use of grey images disregards information about the differences of hue and saturation that may be valuable in image classification. Discriminant analysis was also used in the classification of images. Shuttleworth *et al.* (2002b) followed this up with an application of colour texture analysis to Gaussian smoothed images to measure low frequency texture using co-occurrence matrices. Results showed improved overall classification accuracy. Tjoa and Krishnan in 2002 proposed to obtain quantitative parameters from texture spectra both in the chromatic and achromatic domains using 66 clinically obtained colonoscopic images. Texture spectrum of the RGB components and intensity were obtained from texture unit numbers and six statistical measures (energy, mean, standard deviation, skew, kurtosis, and entropy). There were in total 24 inputs to their algorithms. The features obtained were fed into a supervised back-propagation neural network (BPNN) and to unsupervised neural networks, namely, probabilistic neural network (PNN), learning vector quantization (LVQ), and self organizing map (SOM) for the classification of colonoscopic images. The authors used a faster learning algorithm known as Marquart algorithm to decrease training time. For the BPNN, one-third of the image sample sets were used for training. The training was "online" for the unsupervised networks. The BPNN was able to achieve an overall accuracy of as high as 92.42% while the unsupervised networks achieved a highest accuracy of only

83.33%. Marghani *et al.* (2003) investigated the potential of using morphological analysis based on fractal geometry in classifying colorectal histopathological images. The study made use of the same dataset used by Esgiar *et al.* (1999) of 44 images of normal colon tissue and 58 images of malignant colon tissue. To evaluate the performance of the algorithm, the authors employed analysis of variance or ANOVA. Results indicated that the fractal dimension of cancerous colonic glands was significantly greater than for normal glands. A fuzzy-neural network combined with a clustering algorithm was proposed by Nwoye *et al.* in 2004 to classify cancerous colon cells using fractal dimension techniques and texture features (entropy, correlation, inverse difference moment, and angular second moment). Training and testing images were captured using a light microscope with magnification of x40 and a CCD camera. The study made use of 116 cancerous and 88 normal colon cell images, half of which were used in training, while the other half was used for validation of the algorithm. The authors implemented their algorithm using MATLAB and were able to achieve a classification rate of 96.4%.

Filippas *et al.* (2003a) implemented genetic algorithms (GA) for colonic tissue image classification into normal and cancerous classes on a cluster of Linux workstations using distributed computing techniques. The implementation was reported to have been based on Parallel Virtual Machine or PVM. Three different feature groups were used: features from the image histogram (mean, variance, skewness, kurtosis, entropy and angular second moment); grey-level difference statistic (mean, variance, contrast, entropy and angular second moment); and co-occurrence matrix features (maximum probability, dissimilarity, difference moment, homogeneity, inverse difference moment, entropy and angular second moment). The accuracy for the images from the training set was reported to be 100% while it was 91% for some cases in the test set. Filippas *et al.* (2003b) later compared the classification performances of using GA with using a a supervised backpropagation artificial neural network (ANN). The set

of properties used was similar to the one used previously in Filippas *et al.* (2003a) considering pixel distances of 1, 5 and 9. Two magnification levels were used in the study: 40x and 100x. The training set consisted of 10 images for each case considered (normal, dysplastic, and cancerous) while the test set had less number of images (5 images for normal, 6 images for dysplastic, and 5 images for cancerous). Both the GA and ANN achieved better classification accuracies (as high as 87.5%) for the 40x magnification. Comparison of the two methods was performed through the tabulation of correct classification instances for each class and correct predicted classification of each image to a particular class.

Rajpoot and Rajpoot in 2004 optimised a Support Vector Machine (SVM) classifier for hyperspectral normal and malignant colon tissue cells by finding optimal parameters for three kernel functions: linear, Gaussian, and polynomial. A classification accuracy of over 99% was said to have been achieved using optimal parameters for the Gaussian kernel on a limited data set using multiscale morphological features. A few years later, Rajpoot *et al.* (2006) reported to have used again SVM to classify hyperspectral colon tissue cells using Principal Component Analysis (PCA) and Independent Component Analysis (ICA) to reduce the number of data dimensions . Comparison was made between the use of wavelet based segmentation with PCA and spectral analysis – ICA based segmentation. The segmentation was performed in an unsupervised way through the use of a nearest centroid clustering algorithm. Statistical and morphological categories of features were extracted at multiple resolutions. The statistical features used were: geometric mean, harmonic mean, arithmetic mean, median, trimmed mean, standard deviation, variance, coefficient of variation, second moment, mean absolute deviation, kurtosis, and skewness. The morphological features that were calculated were: area, eccentricity, equivalent diameter, Euler number, extent, orientation, solidity, major axis length, and minor axis length. The results showed that morphological features performed better than the statistical features. The final simulation resulted in

89% sensitivity, 85% specificity, and 87% classification accuracy. Another related study was conducted by Masood and Rajpoot (2006) wherein ICA and k-means clustering were used to accomplish dimensionality reduction and tissue segmentation in the classification of hyperspectral colon tissue images. Morphological features as well as features from grey-level co-occurrence matrices (energy, inertia, and local homogeneity) were used as features. In the classification stage, Linear Discriminant Analysis or LDA was compared with SVM using a 3rd degree polynomial kernel. A comparative study of two classification approaches based on 2D spatial analysis (SA) on a single hyperspectral band and 3D spectral spatial analysis (SSA) carried out in 2004 by Rajpoot and Rajpoot was reported by Masood and Rajpoot (2008). Using 2D principal component analysis (2DPCA) with nearest neighbour classification and circular local binary pattern (CLBP) features with classification techniques used in their earlier work on SSA, it was determined that the approach using SA generated better results compared to SSA. Masood and Rajpoot elaborated more on their SA approach in 2009 reporting a 90% classification accuracy using CLBP features to distinguish the benign and malignant patterns.

Fiscor *et al.* (2008) analysed hematoxylin-and-eosin-stained images for automatic classification as normal mucosa (24 cases), aspecific colitis (11 cases), ulcerative colitis (25 cases), and Crohn's disease (9 cases) using digital slides and virtual microscopy. It was reported that 38 cytometric parameters based on morphometry were determined on cells, glands, and superficial epithelia. The ratio of cell number in glands and in the whole slide, biopsy/gland surface ratio, was found to be the most discriminatory parameter. Leave-one-out discriminant analysis resulted in 88% overall classification accuracy. Gan Lim *et al.* (2010a) used Kohonen Self-Organising Map (KSOM) and grey-level co-occurrence matrix, or GLCM, textural properties to classify colonic histopathological images. The properties that were used were inverse difference moment (IDM), correlation, and the 2 information measures of correlation

(f12 and f13). The U-matrix of the KSOM showed good clustering of the normal cases. LVQ or learning vector quantisation and nearest neighbour algorithm were employed in the classification of the colonic images. Results obtained were preliminary and suggested the use of other feature sets. Gan Lim *et al.* (2010b) proposed the use of average pixel intensity and the presence of circular formations as discriminating features to distinguish between normal and abnormal microscopic colonic images. The use of average pixel intensity was aimed at representing hyperchromasia in abnormal samples while circular formations in normal images accounted for macroarchitectural order. The circular formations in images were measured by implementing a Hough Transform to detect circles from binarised images using Canny edge detection. Using the images from the test set and the average pixel intensity as a feature, all 10 normal images were classified correctly, while only 2 of the 10 adenomatous polyp images were misclassified as normal, and only 3 of the 10 cancerous images were misclassified as adenomatous polyp. No normal images were misclassified as cancerous and vice versa. The use of the Hough Transform to detect circular formations in sample images was tested in a different way. Instead of producing a confusion matrix, a clustering of data points was presented with the variance and the range of the Hough transform accumulator space votes as coordinate axes. The plot of the various data points belonging to normal, adenomatous polyp, and cancerous cases was reported to exhibit excellent clustering. The adenomatous polyp region was verified to be in the mid range between the normal and cancerous regions.

This research was inspired by the well-known potential of combining fuzzy logic (FL), artificial neural networks (NN), and genetic algorithms (GA). One special feature of these three approaches is that they are all derived from or based on nature, specifically biology. Fuzzy logic allows us to deal with imprecise quantities such as ‘low’, ‘medium’, or ‘a bit cold’, which are ‘human’ observations. The fuzzy logic framework gives us a form of mathematics that is able to process vague information,

thus allowing computers to solve problems in a more 'human' way. Artificial neural networks operate by exploiting the power of multiple nodes that are interconnected, much like the neurons in the nervous system, to produce the desired outputs or to identify hidden patterns in the input. Genetic algorithms on the other hand are optimisation algorithms that try to find the best solution among a population of solutions to an optimisation problem. The search for the optimum solution in GA is implemented by allowing populations of candidate solutions to undergo a simplified version of the natural evolution process. In medical images analysis, despite the natural tendency of humans to exhibit variation in performance or output, computers are still considered far inferior from being relied upon in making final diagnostic decisions. It is therefore wise to aim for automated image classifiers equipped with capabilities derived or based on nature or capabilities derived from human beings. It is on this basis that this research was started.

1.2 Statement of the Problem

Basically, the problems that have been addressed in this research are related to the analysis of images wherein the distinction between the subject and background is very difficult if not impossible to make. One of the most important considerations made is the method of comparison between the classification performances of the human pathologists and the computer running the algorithms. This step was not straightforward since human pathologists and computer classifiers utilise images in different ways. In practice, pathologists make judgements about a microscopic sample using a variety of ways or steps and based on a number of factors, most of which must be based on professional experience. Normally, a patient's tissue sample is examined by looking at the entire slide with the pathologist having the liberty to move the

microscope lenses all around the slide sample and also adjust the magnification. In other words, the conventional procedure is that during examination, the pathologist examines the entire slide of the tissue sample. On the other hand, the images that were gathered in this research were all snap shots of specific regions of tissue slide samples and no attention was paid to tagging each image so that one would later know which slide a particular image came from. All that was provided by the collaborating pathologist was a set of colonic microscopic images of different regions from several slides with three classifications: normal, adenomatous polyp, and cancerous. This meant that, in order to conduct a "fair" classification performance comparison, the human pathologists had to be requested to base their assessment on a per image basis. Another limitation in this study is the absence of colour processing of the input images, so the classifier algorithms were trained with grey images only. This particular aspect of the study can be viewed as a constraint of the classifier system. The "right" image size was also an important consideration. An image that is too large might have as much detail as can be calculated but the speed as to what rate the classification process can be implemented might be too time-consuming. An image that is too small might not contain enough information that will render any classifier system totally ineffective. A balance therefore between image size and image processing speed is of absolute importance.

Feature selection process is another important consideration in this study. The scope of this study was confined to the use the Haralick texture features as input to the classifier systems. Haralick *et al.* (1973) suggested 14 features derivable from a grey-level co-occurrence matrix or GLCM. They suggested a feature selection procedure prior to classification as some of the features highlighted are strongly correlated with each other. This is one of the main issues in this study, i.e., how to select the features. Closely related to this problem is the problem of designing or conceptualising the classifier system architecture. Naturally, the choice of specific members of a feature set

will affect the performance of any classifier system regardless of design. What kind of hybrid combination of fuzzy system, neural network, and genetic algorithm is best suited to address these issues?

The range of issues addressed in this study also included the method of measuring the success and failure of a classifier under consideration. Currently, there is no widely accepted performance method or metric similar to the Receiver Operating Characteristic or ROC analysis for N-class, where N>2, classifier (Patel and Markey, 2005). It is believed in this study that the commonly used confusion matrix, together with accuracy percentage, does not provide enough information regarding the faults of a classifier and is in effect part of the problem. The accuracy percentage which is normally computed by normalising the main diagonal of the confusion matrix will always give the same result regardless of whether the misclassification was from, say, cancerous to polyp or from cancerous to normal. Cleary a misclassification from cancerous to normal is much more severe than misclassification from cancerous to polyp. A novel metric of quantifying the performance of a classifier system is therefore needed; something that will provide opportunity for improvement based on the gravity of failure or misclassification.

1.3 The Proposed Solution

As discussed in section 1.2, the problems in this study can be enumerated as follows:

- comparison of classification performances between human pathologists and a computer running the classifier algorithms,
- feature selection process,

- classifier system architecture design, and
- a need for a better and more useful classifier performance metric.

To tackle the first of the problems enumerated, it was decided to simply come up with survey forms with printed monochromatic pictures of some of the sample images for pathologists to classify into normal, adenomatous polyp, and cancerous using their professional e4xperiences. Pathologists are busy people and therefore it was not practical to request them to try to classify all of the 90 test images. After a trial survey with some pathologists, it became clear that a 300 x 400-pixel image size printed on A4 bond paper was already acceptable as there were no complaints regarding the printed image size from the participants themselves. This was how the pathologist survey forms were designed: 3 monochromatic images per sheet of A4 bond paper with each image having dimensions of approximately $3\,{}^{1}/_{8}$ inches by $4\,{}^{3}/_{16}$ inches.

The feature selection process problem was addressed by exploring two methods. The first method utilised a variance ratio first used by Boland *et al.* (1998). This variance ratio is a modified version of the Multiple Discriminant Analysis or MDA wherein the between-class variance for every candidate feature is normalised by the sum of the within-class variances. Features with high variance ratios are considered as exhibiting good clustering attributes. A desirable characteristic of the variance ratio is that it allows one to search for features that widely separate the different classes and simultaneously group together similar elements into clusters. The other method that was used in selecting the features involved genetic algorithm (GA) and Kohonen Self-Organising Map (KSOM). The GA was used to search for combinations of features that produce minimal KSOM training map error. The input to the GA was a set coefficients while the input to the KSOM was the set of all the features, each multiplied by a corresponding coefficient input to the GA. Features were chosen based on the resulting

coefficients after the application of the GA operators on several populations of coefficients.

Although there are many other image properties that can be considered to be part of the property selection set to choose from, focus was made on the texture properties derivable from grey-level co-occurrence matrices (GLCM) introduced by Haralick *et al.* (1973). The reason for doing this is that previous studies such as the ones reported by Esgiar *et al.* (1999), Atlamazoglou *et al.* (2001), Shuttleworth *et al.* (2002a, 2002b, 2005), and Nwoye *et al.* (2004), to name a few, have shown that texture information from GLCM is very useful. Morphological image analysis was also considered. However, the nature of the images used in this study is one of irregular and complicated structural shapes and doing so might shift the research focus mainly to candidate feature set selection. The main objective of this study is to investigate the development of a hybrid classifier system. The distance used in calculating all the GLCMs was 1 pixel, based on the suggestion by Zucker and Terzopoulus (1980) to optimise GLCM by maximising chi-square significance test. Investigation using other pixel distance values was therefore not given priority.

The classifier system design and the kind of inputs seemed well suited for something that combines the power of fuzzy logic and artificial neural networks. One of the modern most powerful computational tools available to the scientific community is the ANFIS which is short for Adaptive Network-Based Fuzzy Inference System.

To address the need to devise a new metric for classifier system performance, Mean Relative Difference Confusion Matrix (MRDCM) and Classification Performance Index (CPI) are being proposed in this study. MRDCM is a matrix much like the conventional confusion matrix, except that the elements in an MRDCM are differences between the ANFIS classifier output and each element of the vector $[0.0 \quad 0.5 \quad 1.0]^T$. The ANFIS classifier had been trained using 0.0 to denote normal, 0.5 for adenomatoous polyp, and 1.0 for cancerous classification. Classification based on real

numbers might be more useful sometimes since an output that tells about the relative location of a particular case within the spectrum of possible cases surely contains more information. The introduction of the CPI measure is an attempt to summarise classifier performance in a single number rather than through the use of a matrix of numbers. The CPI is calculated by algebraically adding the rewards of classification and penalties of misclassification committed by a classifier. Different levels of misclassifications are given different penalties. This scheme therefore allows for the distinction between classifiers that have an equal number of correct classifications but have different kinds of misclassifications. The objective of CPI is to penalise more severely 2 levels of 'downgraded' misclassifications, *e.g.* cancerous misclassified as normal.

1.4 Aims of the Study

The challenges in this study are not unique to the area of automated colonic image cancer detection. These are also faced by researchers investigating other types of cancer using medical images, *e.g.* breast mammography, blood image analysis, colonoscopy, to name a few. However, this study proposes to apply hybrid algorithms that combine the advantages of fuzzy logic, neural networks, and genetic algorithms to solve the problem of image classification specifically in the area of colonic cancer detection using textural features.

The overall aim of this study is to develop and evaluate efficient hybrid algorithms that use neural networks, fuzzy logic, and genetic algorithm paradigms to automatically identify colonic histopathological images into normal, adenomatous polyp, and cancerous classifications.

The following are the specific objectives:

1.4.1 To develop hybrid classifier algorithms to distinguish images of dysplastic and cancerous colonic mucosa from normal ones.

1.4.2 To evaluate the effectiveness of the algorithms to be developed by using a subset of images not included in the training phase and minimise the classification error.

1.4.3 To compare the algorithm performance in a clinical setting against consultant histopathologists' expert classifications.

1.4.4 To further refine the final hybrid structure and undertake further tests to ensure its robustness under clinical conditions.

1.5 Novel Contributions to the State of the Art

The novel ideas and contributions of this thesis to scientific knowledge can be summarised into the following:

1. The evaluation of feature sets using GA-KSOM;
2. The use of ANFIS to classify colonic histopathological images;
3. The introduction of the Relative Mean Difference Confusion Matrix (MRDCM), an effective assessment tool for ANFIS classifier;
4. The introduction of the Classification Performance (CPI) parameter, a better measure of classifier performance derived from a conventional confusion matrix; and
5. The presentation of the results of a mini assessment survey of classification skills of human pathologists in Manila, Philippines.

The combination of GA and KSOM as proposed and utilised in this study is a novel approach to feature selection, particularly as applied to feature set identification in colonic image analysis. It is an attempt to combine two natural processes: evolution

and (unsupervised) learning. In a way, this procedure uses nature itself to search for the solution/s to the problem of feature selection in this study. This particular scheme has never been examined before in previous studies, and especially in relation to colonic image analysis. The most straight forward method in selecting a feature set is simply to let a human 'expert' select features heuristically or perhaps intuitively. This method however lacks a solid theoretical basis and is therefore characterised by arbitrariness in its success. This means that a search for a better alternative is clearly needed. Rajpoot *et al.* (2006) used Principal Component Analysis, or PCA, to deal with multiple numbers of features. The use of PCA however is not desirable since it only transforms the feature space and does not reduce the number of features to be extracted from the images. In other words, PCA does not implement feature selection. In 2003, Filippas *et al.* used GA and a feed-forward artificial neural network (ANN) to classify colonic images. Unlike in the method used in this study, the GA and the ANN were applied separately and results were compared subsequently. Also, Filippas *et al.* (2003) used a supervised feed-forward ANN which is different from KSOM. The use of KSOM in this study instead of a feed-forward ANN allows for the avoidance of supervised training while carrying out the process of feature selection. The most important advantage in using KSOM over the feed-forward ANN is that the classification in the training data is not necessary. This can prove to be useful especially when there might be errors or inaccuracy in the classification of the training data. This approach can shield the process of selecting 'good' properties from the biases that the expert pathologist might have had in producing the classification in the training data.

The application of ANFIS to the classification of colonic histopathological images is another novel contribution of this study. To the author's knowledge, ANFIS has never been applied to colonic image classification in previous studies. Tjoa and Krishnan in 2002 applied feed-forward ANN, probabilistic neural network (PNN), learning vector

quantization (LVQ), and self organizing map (SOM) for the classification of colonoscopic images. Nwoye et al. (2004) proposed a fuzzy-neural network combined with a clustering algorithm. The architecture proposed was different from ANFIS. Furthermore, as mentioned in the previous paragraph, Filippas et al. (2003b) used GA and ANN to classify colonic images. All such approaches are clearly different from ANFIS. Part of the contribution of applying ANFIS in this study is the fact that the single output of the ANFIS architecture was not seen as a burden or problem, considering that there are three (3) output classes, but rather as an advantage. Knowing that colonic images can be characterised by a spectrum of conditions, from normal to cancerous, makes ANFIS very much suitable to the problem in this study since the output can be made to range from 0.0 to 1.0. This range is ideal in representing cases that are dysplastic with varying degrees of abnormality. Similar to a multilayer perceptron, ANFIS is also capable of learning from a set of training data. However as pointed out by Jang (1993), ANFIS has certain advantages over the multilayer perceptron. In addition, ANFIS also discovers and generates “knowledge” from the training data set in the form of fuzzy rules and membership functions.

Related to the use of ANFIS in this study is the Relative Mean Difference Confusion Matrix (MRDCM) which is a novel method of accounting for the performance of a classifier. Since the natural output of an ANFIS classifier is a range of real numbers, the usual confusion matrix could not be utilised unless threshold values were selected to categorise the output value into normal (N), adenomatous polyp (P), and cancerous (C). The chosen values to represent each of the ideal N, P, and C cases were 0.0, 0.5, and 1.0 respectively. The real numbers 0.0 and 1.0 were chosen since they represent extreme values in the same way as N and C cases do in characterising colonic images. Since dysplastic or adenomatous polyp (P) cases are considered to be somewhere in the middle of the N and C cases, a 0.5 value was chosen to represent cases belonging to the P classification. The MRDCM simply presents the average distances of the test

images from the three values mentioned for each of the cases. Unlike in a conventional confusion matrix, an ideal classifier is expected to have zeros or small values in the main diagonal of an MRDCM. Together with an ANFIS classifier without thresholds, the MRDCM proposes an alternative way of analysing and evaluating a colonic image classifier. The idea is to disregard the specific classes and rather focus on the relative position of a classification output in the assumed range of the classification spectrum. This concept allows one to do away with threshold values in the output of a classifier that outputs a range of real numbers. The threshold values can sometimes render the classification process ineffective if for instance the output is meant to be interpreted in a fuzzy way.

The coefficient of performance, or CPI, parameter carries the idea behind the (percentage) classification accuracy a step further by introducing factors that can account for misclassifications of the classifier in question. Since the classification accuracy parameter only considers the correct classifications that were made, two classifiers, wherein, for example one has misclassified a cancerous case as normal while the other has misclassified a cancerous case as adenomatous polyp, might be evaluated as having performed equally. This is clearly not how humans would evaluate classifier performance. If one is attempting to improve the performance of a classifier or attempting to select the better classifier, the gravity of mistakes committed must be taken into consideration. The CPI parameter overcomes this problem. When one computes a CPI value, each element of the confusion matrix is given a multiplying factor proportionate to its 'importance', which is not the case with the conventional classification accuracy parameter.

The comparison between the classification performances between the ANFIS implementations and Philippine pathologists is very insightful. First, it suggests validation of the effectiveness of the ANFIS classifiers developed in this study since the results show similar trends. Second, the comparison confirms that human pathologists

make mistakes and therefore it is quite possible for some of the misclassifications of the ANFIS classifiers not to be real 'mistakes' but rather more as a disagreement in 'professional' judgement. This is seen as an important contribution to the results of this research. With regard to the difference in training times and methods between the pathologists and the ANFIS implementations, it is fair to say that the pathologists had the upper hand since humans naturally have a much more advanced vision analysis system and have been trained for several years in medical school and professional practice as compared to the limited training time of the algorithms. Therefore the success of the ANFIS implementations cannot be seen as resulting from an unfair comparison in its favour.

1.6 Structure of the Thesis

This thesis contains 6 chapters in total. The Reference and the Appendices sections have been placed after the final chapter.

Chapter 1 – The 1st chapter is an introductory chapter and contains discussions regarding the background and origins of the research, the research problem, the proposed solution, the research aims and the novel contributions made by this study to the state of the art.

Chapter 2 – The 2nd chapter provides a short introduction to Colon Cancer. Things such as staging and grading systems and survival rates for colon cancer are briefly discussed here.

Chapter 3 – The 3rd chapter presents a summary of some of the fundamental image analysis tools/algorithms and techniques used in the research such as histogram equalisation, scatter matrices, image texture, KSOM, ANFIS and GA. A short

specification list of the hardware used in the implementation of the algorithms is placed at the end of this chapter.

The next two chapters, chapters 4 and 5, contain the main body of the thesis. In these chapters, the detailed information on the implementation of the algorithms can be found.

Chapter 4 – Chapter 4 is focused on the feature selection processes using ratio of variances and GA with KSOM.

Chapter 5 – The 5th chapter discusses the implementation of image classification using ANFIS and the feature sets suggested in Chapter 4. Novel metrics for classifier performances are also introduced here. Finally, results of a survey conducted on a few human pathologists regarding their abilities to classify colonic histopathologic images are reported and compared with the performances of the algorithms developed in this research.

Chapter 6 – The final chapter concludes this thesis based on the findings and accomplishments made in this research and offers recommendation for future work.

Chapter 2 - WHAT IS COLON CANCER?

According to the World Health Organization (2009), colon cancer is considered the third leading cause of cancer mortality in the world with an estimated 639,000 deaths each year. In the UK, colon cancer is the third most common cancer with around 16,000 deaths out of 36,500 people diagnosed each year (Cancer Research UK, 2008a). Ngelangel and Wang reported in 2002 that colon and rectum cancer is among the leading cancer types in the Philippines. The majority of people with this type of cancer belong to the older population with 80% of cases found to be in those over 60 years of age (Dorundi and Bannerjea, 2006). Other names used for colon cancer are bowel cancer and colorectal cancer.

Cancer is generally understood to be a case of an uncontrolled growth of a cell or group of cells which tends to invade adjacent tissues and spread to other parts of the body (metastasis). In medical terms, cancer is usually referred to as malignant neoplasm or tumour. While the words neoplasm and tumour both mean abnormal cell growth, not all cancers form tumours. An example of a non-tumour-forming cancer is leukaemia. The invasive metastatic nature of cancer cells is the major cause of death from cancer. Cancer cells have a number of histopathological characteristics. Figure 2.1 illustrates and outlines the basic differences between normal and cancerous cells.

This image has been removed

Figure 2.1 Structural differences between normal and cancerous cells (National Cancer Institute, 2008a)

It is believed that cancer is a result of a disorder in the mechanism by which cells repair their DNA. It is not yet fully understood why some people get cancer and others do not; however, many experts think that some factors are more important to consider than others. The National Cancer Institute or NCI groups these factors into two: intrinsic factors and extrinsic factors. Intrinsic factors include heredity, diet, and hormones, while extrinsic factors include radiation, some chemicals, and some viruses and bacteria (National Cancer Institute, 2008b). Genetic mutation plays a big role in cancer formation. It is understood that cancer-causing agents can sometimes cause some genes to mutate and enable the affected cells to multiply uncontrollably and invade healthy cells. Cancer-causing agents can cause proto-oncogenes to become oncogenes as illustrated in Figure 2.2. Proto-oncogenes are genes which are responsible for the regulation of cell growth and differentiation while oncogenes, on the other hand, promote hyperactive cell growth and division, resistance to cell death, and invasion of other parts of the tissue or body. When proto-oncogenes become

oncogenes, the normal production of cells ceases. Aside from proto-oncogenes and oncogenes, there are other classes of genes that are involved in cancer formation or prevention. The human body has a number of genes that can prevent abnormal cell growths. Genes known as tumour suppressor genes can counter the effects of excessive cell proliferation and are sometimes called anti-oncogenes. Another class of genes known as suicide genes can order cells that have been damaged severely to commit suicide or die naturally (apoptosis) thus preventing the reproduction of cells with altered DNA. Possible errors in the DNA duplication during cell division can be corrected by DNA-repair genes. During mutation, however, it is possible that these natural defences of the body can be inactivated and thus allow a series of events that can eventually lead to cancer.

Figure 2.2 Mutation of proto-oncogene into oncogene
(National Cancer Institute, 2008c)

Cancers are usually categorized based on the tissue type of origin of the cancerous growth as enumerated below:

1. Carcinomas
2. Sarcomas

3. Lymphomas
4. Leukemias.

There are several types of cancer (see Figure 2.3). Carcinomas are probably the most common types of cancer. Cancers under this type arise from the cells that cover external and internal body surfaces. Sarcomas arise from cells belonging to the supporting tissues of the body e.g., bone, cartilage, fat, connective tissue and muscle. Cancers that arise in the lymph nodes and tissues of the body's immune system are called lymphomas, while leukaemias are cancers that involve the blood cells and the bone marrow. Usually, cancers are named based on the organ or type of cell in which the cancerous growth originate. Colon cancer, for example, involves cancerous growths in the large intestine or the colon, the rectum, and the appendix. Through metastasis, cancer can spread to other organs and can eventually result into death of the patient if not treated successfully.

Figure 2.3 Different types of cancer
(National Cancer Institute, 2008d)

These images have been removed

Figure 2.4 Normal and Cancer Cell Division
(National Cancer Institute, 2008e)

Figure 2.4 shows diagrammatically the progression of normal to cancer cells. Progression from normal to cancer involves hyperplasia and dysplasia as in-between cases. Hyperplasia differs from dysplasia by the nature of the cells involved in the abnormal growth. Unlike dysplastic cells, hyperplastic cells are still responsive to normal regulatory control mechanisms of the body. A precursor to cancer involving epithelial cells is known as carcinoma in situ or CIS. A CIS lesion is characterised by an absence of invasion of the surrounding tissue. Severe dysplasia and carcinoma in situ are considered to mean practically the same thing. Some CIS do turn into tumour and are therefore recommended to be removed completely by medical doctors. Benign tumours that have glandular origins are called adenomas. Adenocarcinomas are adenomas that have turned into cancer.

As stated, colon cancer is a cancer that involves the large intestines, the rectum, and the appendix. Generally, colon cancer is a disease of older people with almost 75% of cases in people aged 65 and over (Cancer Research UK, 2008b). Figure 2.5 shows some statistical information about bowel cancer in the UK for 2005. It is generally accepted that high intake of red meat and processed meat and low intake of fruits and vegetables tend to increase the risk of developing colon cancer. 'Westernisation' of lifestyle and diet has been linked to an increase in the risk of colon cancer incidence. Research suggests that environmental factors play a major part in the aetiology of the disease. People who have migrated to a new place or country and have adapted to the lifestyle of the people in that place have been observed to also acquire the risk associated in that area. As an example, the risk of getting colon cancer for offspring of Japanese migrants to the United States is three or four times higher than among the Japanese in Japan (Boyle and Langman, 2000). Physical inactivity, being overweight, alcohol consumption and heredity have also been linked to an increased risk of getting colon cancer. Incidence for males is higher than for females for ages above 40.

This graph has been removed

Figure 2.5 New bowel cancer cases and age-specific incidence rates by sex in the UK for 2005 (Cancer Research UK, 2008c)

An important part of cancer treatment and research is known as staging. The American Cancer Society (2008) defines cancer staging as the process of determining through medical tests how far cancer has spread. The current most accepted staging system for colorectal cancer is the TNM System, developed and maintained by the American Joint Committee on Cancer (AJCC) and the International Union Against Cancer (UICC). TNM stands for tumour, nodes, and metastases. Basically, the TNM system is aimed at describing the extent of the tumour, the extent of spread to the lymph nodes, and the presence of metastasis. The T, N, and M categories of a patient are usually combined in order to summarise the information in what is called stage grouping. There are eight (8) AJCC stage groupings: stage 0, stage I, stage IIA, stage IIB, stage IIIA, stage IIIB, stage IIIC, and finally, stage IV (see Table 2.2). Figures 2.6 up to 2.10 can be used as guide illustrations to have a clear picture of the various parts mentioned in the TNM staging system in Table 2.1.

Table 2.1 Summary of the TNM staging system

T category	N category	M category
TX - Primary tumor cannot be evaluated	Nx - Regional lymph nodes cannot be evaluated	MX - Distant metastasis cannot be evaluated
T0 – no evidence of primary tumour	N0 - No lymph node involvement is found.	M0 - No distant metastasis (cancer has not spread to other parts of the body)
Tis - Carcinoma in situ (early cancer that has not spread to neighboring tissue)	N1 - Cancer cells found in 1 to 3 nearby lymph nodes.	M1 - Distant metastasis (cancer has spread to distant parts of the body)
T1 - cancer invasion through submucosa into lamina propria	N2 - Cancer cells found in 4 or more nearby lymph nodes.	
T2 – cancer invasion into the muscularis propria (outer muscle layer)		
T3 – cancer invasion into the subserosa but not to any neighboring organs or tissues.		
T4 - cancer through the wall of the colon or rectum and into nearby tissues or organs.		

Table 2.2 AJCC Stage Groupings

Stage Grouping	TNM staging
Stage 0	Tis, N0, M0
Stage I	T1, N0, M0 or T2, N0, M0
Stage IIA	T3, N0, M0
Stage IIB	T4, N0, M0
Stage IIIA	T1, N1, M0 or T2, N1, M0
Stage IIIB	T3, N1, M0 or T4, N1, M0
Stage IIIC	Any T, N2, M0
Stage IV	Any T, Any N, M1

Figure 2.6 Cross-section of the Colon (National Cancer Institute, 2008f)

These images have been removed

Figure 2.7 T1 tumour invasion of the colonic tissue (Greene *et al.* 2006: 111)

Figure 2.8 T2 tumour invasion of the colonic tissue (Greene *et al.* 2006: 111)

These images have been removed

Figure 2.9 T3 tumour invasion of the colonic tissue (Greene *et al.* 2006: 112)

This image has been removed

Figure 2.10 T4 tumour invasion of the colonic tissue (Greene *et al.* 2006: 113)

Another staging system that is important to mention is the Duke's system. It is a classification system that has been replaced by the TNM system but is still used by many physicians. It is much simpler than the TNM system since it only uses the first four (4) uppercase letters of the English alphabet to identify the cancer stages. The Duke's system is outlined on the Table 2.3.

Table 2.3 Duke's Staging System for Colorectal Cancer

A	the tumour is confined to the intestinal wall
B	the tumour is invading through the intestinal wall
C	there is already lymph node involvement
D	with distant metastasis

Table 2.4 Survival rates for colon cancer by stage (American Cancer Society, 2008).

Stage	Survival Rate
Stage I	93%
Stage IIA	85%
Stage IIB	72%
Stage IIIA	83%
Stage IIIB	64%
Stage IIIC	44%
Stage IV	8%

The survival rate generally decreases with stage. As shown in Table 2.4, since the survival rate is generally inversely correlated with the stage number, early detection and treatment of cancerous growths is of paramount importance in patient survival. The reasons why the survival rate data for Stage IIIA is higher than that of Stage IIB is unclear according to the American Cancer Society (2008), however the trend is obvious.

In addition to staging, another important tool that health experts use in cancer treatment and research is histologic grading. According to the AJCC Cancer Staging Atlas (Greene *et al.*, 2006), histologic grade is the qualitative assessment of tumour differentiation in terms of the resemblance of the tumour itself with the normal tissue at that site. The National Cancer Institute (2008g) defines differentiation as a measure of how mature or developed cancer cells are in a tumour. Undifferentiated or poorly differentiated tumour cells tend to lack the structure and function of normal cells and grow uncontrollably. On the other hand, differentiated tumour cells appear to be similar to normal cells and tend to grow and spread at a slower rate. Grading systems enable experts to classify neoplasms in terms of microscopic appearance of the cells involved and make histopathologic assessment. The widely accepted AJCC grading system is summarised in Table 2.5.

Table 2.5 AJCC cancer grading system

GX	**Grade cannot be assessed**
G1	Well differentiated (Low grade)
G2	Moderately differentiated (Intermediate grade)
G3	Poorly differentiated (High grade)
G4	Undifferentiated (High grade)

Various observations indicate that more than 70% of colon cancer cases arise from adenomas in the colon, more commonly known as adenomatous polyps (Dorundi and Bannerjea, 2006). Based on this observation, it is widely believed that the removal of

polyps in the colon can significantly reduce the occurrence of colon cancer. The concept that most carcinomas in the colon and rectum arise from adenomas is known as the '*adenoma-carcinoma sequence*'. Figures 2.11 and 2.12 illustrate how it might be possible for tumours to cause some form of obstruction in the colonic lumen and can possibly cause one of the symptoms of colon cancer which is the feeling of incomplete defecation and reduction in stool diameter. The likelihood of the adenoma-carcinoma sequence increases with adenoma size and volume of villous tissue.

Figure 2.11 Appearance of polyps in the colon (National Cancer Institute, 2008h)

These images have been removed

Figure 2.12 Colorectal Cancer Staging (National Cancer Institute, 2008i)

Colon cancer is considered a preventable disease having a slow progression from pre-malignant to cancerous conditions. As such, it satisfies many of the WHO criteria for population cancer screening (Dorundi and Bannerjea, 2006). The general aim of population cancer screening is to detect a disease at an early stage, thereby increasing the chances for patient recovery and/or survival if the appropriate treatment is immediately started. In the UK, the National Health Service (NHS) Bowel Cancer Screening Programme has already been rolled out across the country. Regular screening for bowel cancer has been shown to reduce the risk of death from bowel cancer by 16% (NHS, 2008). The NHS screening programme offers screening every 2 years to all men and women aged 60-69 with people over 70 given screening kits only upon request. The tests that are included in the programme are the faecal occult blood (FOB) test and colonoscopy. The FOB test is an initial and standard test for everyone undergoing the screening process while the colonoscopy is usually only administered to those individuals who demonstrate abnormal FOB test results.

Chapter 3 – REVIEW OF IMAGE ANALYSIS AND ALGORITHMS USED

This chapter summarises some of the important image analysis tools and algorithms that were employed during the implementation of the ideas in this research. The aim of this chapter is to provide a sort of 'bridge' between the proposed solutions devised in this study and the basic materials which can be found on Image Analysis and Soft Computing textbooks. This is by no means an attempt to compile something that will serve as an introductory material to the topics outlined in this chapter.

3.1 Digital Image

A digital image is a 2-dimensional discrete function representing measures of brightness at various points, called pixels, given by a set of coordinates. An image can be either binary, grey, or colour. A grey image contains only a single matrix of numbers with each element giving a particular brightness intensity value within the spectrum from black to white. The term 'sampling' is used to refer to the resolution of the spatial coordinates of an image. The resolution of the brightness in each pixel is called 'quantisation'. Usually, a grey image has 256 levels of quantisation. A binary image is similar to a grey image with only 2 levels of quantisation – 0 for black and 1 for white. Colour images can be thought of as a combination of grey images. For example, an RGB colour image is composed of 3 grey images: 1 for red intensity, 1 for green intensity, and another for blue intensity. The size of a digital image is normally given as M x N where M is the number of rows and N is the number of columns. The convention is to base the coordinates at the upper left corner of an image similar to how elements

of a 2D matrix are sequentially arranged. Figure 3.1 illustrates this coordinate convention.

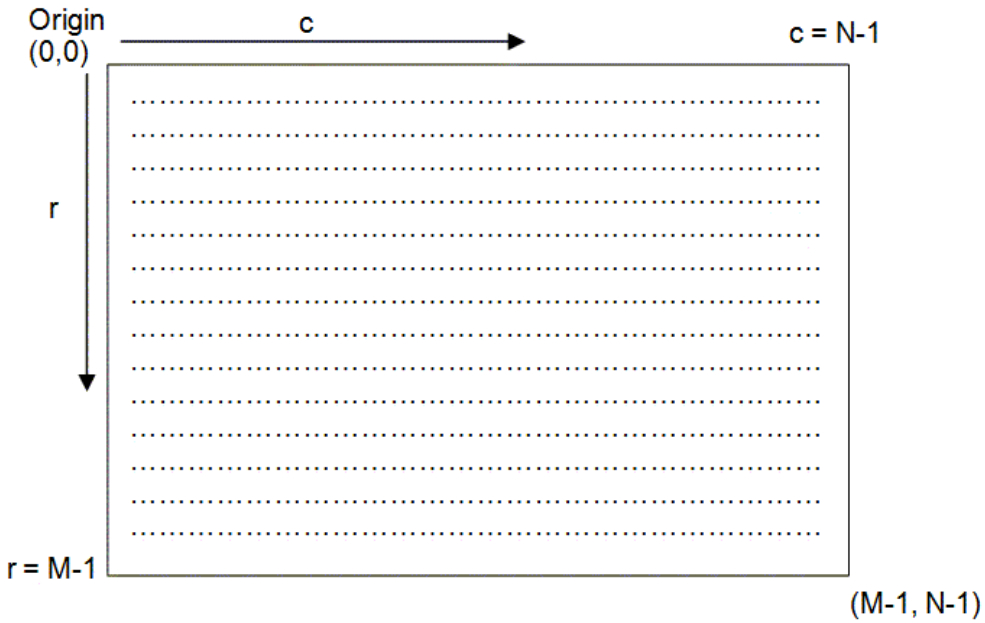


Figure 3.1 Coordinate convention for a digital image

In Figure 3.1, if one needs to refer to the pixel which is, say, 23 pixels from the pixels on the top edge and, say, 37 pixels from of the left-most pixels of the image, the coordinates could be specified as P(23,37) assuming that P is a 2-dimensional array that holds the image data. Take note that in some textbooks on Image Processing, the origin, which is the pixel at the upper-left-most part of an image, is designated with coordinates (0,0). In MATLAB however, the origin is designated as (1,1), therefore in the simple example just mentioned, the same pixel would be referred to as P(24,38). MATLAB is the chosen platform to implement the algorithms in this study.

The images used in this study were all grey images even though they were all captured by a digital camera in colour. This should not be seen as a limitation of the study but as a choice of the author. Shuttleworth *et al.* (2002a, 2002b) reported that the use of colour in texture analysis offered improvement in classification. The improvement however was not really significant and clearly does not suggest that

texture analysis using grey images is obsolete. Moreover in this case, improvements in classification performance using colour information in texture analysis hardly justifies the added complexity that is encountered beyond the use of monochromaticity and the loss of processing speed which are inherent consequences when using colour information. The use of binary images, on the other hand, does not offer enough information in the analysis of the colonic images, more so obviously when trying to use texture information.

3.2 Histogram Equalisation

The histogram of a digital image is a function expressing the frequency of occurrence of each discrete grey level for all the pixels. A normalised histogram is obtained when all the elements of the original histogram are divided by the total number of pixels in the image. From a basic probability concept, the normalised histogram can be interpreted as an estimate of the probability of occurrence of each grey level (Gonzalez *et al.*, 2004). From this point forward, all references to the term 'histogram' imply normalised histogram unless stated otherwise. Histogram equalisation is a transformation process that aims to transform the pixel intensities of an image to create another version of the given image with a more uniform histogram. One of the effects of histogram equalisation is 'automatic' contrast stretching without user intervention or input. Ideally, the transformed image should have a perfectly flat histogram but, because of the discrete nature of a digital image, only an approximation is achieved. To implement histogram equalisation, Equation 3.1 can be used to transform pixel value p to q.

$$q = \frac{L-1}{MN}\sum_{i=0}^{p} O(i) \qquad \textbf{Equation 3.1}$$

where q = new pixel value corresponding to old pixel value p
p = old pixel value
L = number of grey levels or the quantisation
O(i) = cumulative histogram or cumulative distribution function of the image
M = number of rows of the image
N = number of columns of the image

Equation 3.1 basically calculates the so-called cumulative distribution function or CDF of an image and multiplies it by the ratio of the amount of quantisation over the total number of pixels involved. This is best illustrated by an example. Assume that a 4-bit grey scale image has a histogram given as:

grey level i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
number of pixels per grey level, n_i	0	15	0	0	0	0	0	0	0	110	70	80	45	0	40	0.

The numbers given above simply mean that there are basically 16 grey levels, numbered from 0 to 15, since the image is a 4-bit grey image. The second row indicates the number of pixels having the corresponding grey level values given in the first row: 15 pixels have a common grey level value of 1, while 110 pixels have a common grey level value of 9, 70 pixels have a common value of 10 and so forth. The zeroes in the second row indicate that not all grey levels are used, for example, the zero under grey level 15 means that there are no pixels having a grey level value of 15. Graphically, the given histogram can be expressed as in Figure 3.2.

One can notice in Figure 3.2 that, in the given example, the majority of the pixels have ‘high’ values. This means that the image will appear to be bright and therefore has poor contrast. The application of histogram equalisation can redistribute the values of the pixels such that the image will have a “better” contrast. Table 3.1 summarises the implementation of Equation 3.1.

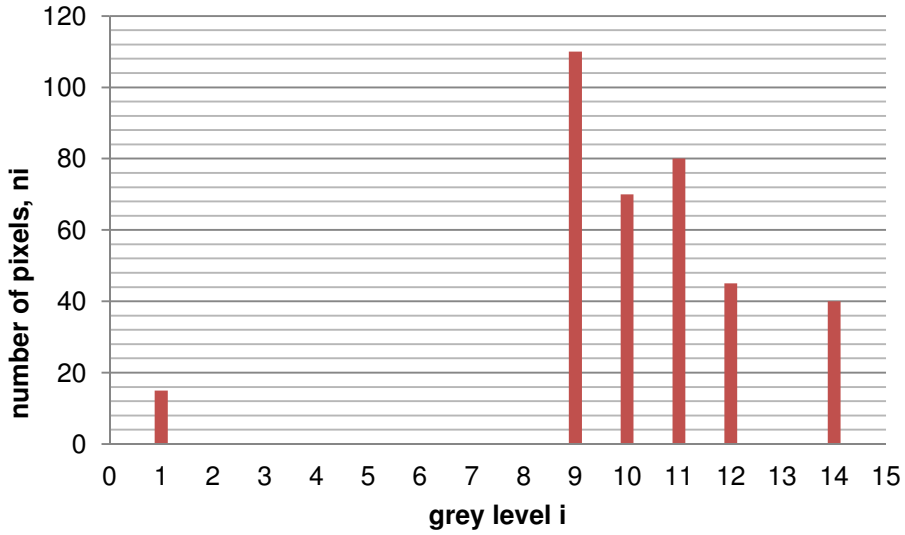


Figure 3.2 Histogram of the given image in the example before histogram equalisation.

Table 3.1 Calculation of the new histogram of the image in the given illustrative example on histogram equalisation.

Grey level	Number of pixles, n_i	Normalised n_i	Cumulative histogram, O(i)	new pixel value	rounded new pixel value	new n_i
0	0	0	0	0	0	0
1	15	0.041667	0.04167	0.625	1	15
2	0	0	0.04167	0.625	1	0
3	0	0	0.04167	0.625	1	0
4	0	0	0.04167	0.625	1	0
5	0	0	0.04167	0.625	1	110
6	0	0	0.04167	0.625	1	0
7	0	0	0.04167	0.625	1	0
8	0	0	0.04167	0.625	1	70
9	110	0.305556	0.34722	5.208333	5	0
10	70	0.194444	0.54167	8.125	8	0
11	80	0.222222	0.76389	11.45833	11	80
12	45	0.125	0.88889	13.33333	13	0
13	0	0	0.88889	13.33333	13	45
14	40	0.111111	1	15	15	0
15	0	0	1	15	15	40
total	n=360	1				n=360

The third column of Table 3.1 is obtained by dividing each entry in column 2, or the 'number of pixels' column, by the total n which happens to be 360. Each time the values in the 3rd column are accumulated starting from grey level 0, the values are listed along the next column, which is column 4 ('cumulative histogram' column). The 5th column, dedicated for the new pixel value, is filled with values by multiplying entries in the 4th column or the 'cumulative histogram' column by L-1 which in this case is 16-1 or simply 15. These values are rounded off to the nearest integer to complete the column for 'rounded pixel value' or the 6th column. The values in this column represent the new values of the corresponding grey level values given in the 1st column. For example, grey levels from 1 to 8 are supposed to be changed to a common value of 1. Grey level 0 will remain as grey level 0. Grey level 9 will now be grey level 5, grey level 10 will become grey level 8, and so on. The 7th column which is the right-most or the last column is the output of the histogram equalisation process. It is obtained by observing the distinct values in the 6th column or the 'rounded pixel value' column. The values in the 2nd column, the 'number of pixels' column, go with the reassignment of the pixel values. For instance grey level 10 in the 1st column with 70 pixels (at the 2nd column) having the same values, these same pixels will have a new grey level which has been calculated to be 8 in the histogram equalised image. This is why $n_i = 70$ has been moved from grey level 10 in the 2nd column to grey level 8 in the last column. The same explanation also applies to the rest of the entries in the last column. This is how the reordering of pixel values is achieved. The new histogram resulting from the histogram equalisation process is shown in Figure 3.3. It can be noticed that the pixel values are no longer concentrated over a single side of the graph. Histogram equalisation is implemented in MATLAB using the command "histeq(f, nlev)" where f is the input image and nlev is the number of intensity levels or quantisation of the image.

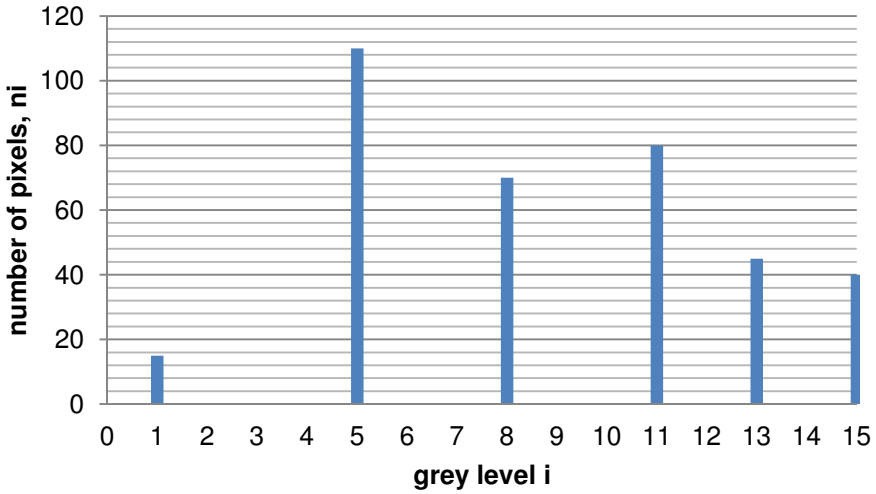


Figure 3.3 Histogram of the given image in the example after histogram equalisation.

Unfortunately in this study, it was found that histogram equalisation process has a tendency to destroy the texture information of an image. This is due to the reordering nature of the process itself in order to improve contrast. Therefore, even though this is an elegant process to improve image contrast, it is not advisable to implement this prior to the extraction of GLCM textural properties.

3.3 Unsharp Masking

Unsharp masking is a spatial filtering technique which can be used to enhance the edges in an image. Sometimes, unsharp masking is called 'edge enhancement' or 'edge crispening' (McAndrew, 2004). The unsharp masking operation is carried out by subtracting a scaled unsharp version of the image from the image itself. The result is an image with enhanced edge pixel values. A more common implementation of unsharp masking is performed by adding to the image a negative of the Laplacian of the original image. In this way, the parameter α of the Laplacian can be used to control the effect on the output image. Figure 3.4 shows the schema of a common implementation of unsharp masking that uses a Laplacian operation.

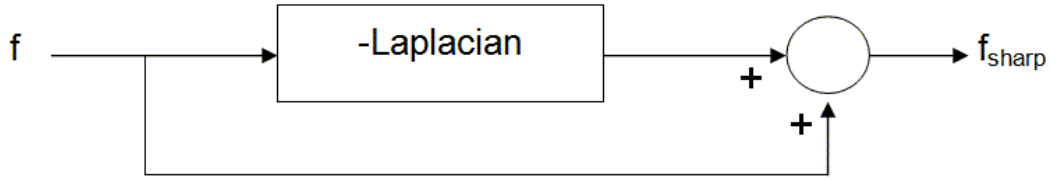


Figure 3.4 Schema of unsharp masking using Laplacian mask

$$\frac{1}{\propto+1}\begin{bmatrix} -\propto & \propto -1 & -\propto \\ \propto -1 & \propto +5 & \propto -1 \\ -\propto & \propto -1 & -\propto \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{1}{\propto+1}\begin{bmatrix} \propto & 1-\propto & \propto \\ 1-\propto & -4 & 1-\propto \\ \propto & 1-\propto & \propto \end{bmatrix}$$

Unsharp mask

Identity Filter

Laplacian mask

Figure 3.5 Relationship between the Laplacian mask and the unsharp mask. α is the Laplacian parameter that controls the effect on the output image. α = 0.5 was used in this study.

Shown in Figure 3.5 is the relationship between the Laplacian mask and the unsharp mask. The Laplacian parameter, α, which varies from > 0.0 to 1.0, controls the effect of unsharp masking to the output image. MATLAB uses α = 0.2 as default value and implements unsharp masking using the commands

m = fspecial('unsharp', α)

and

w = filter2(m, v)

where m is the generated unsharp filter mask, alpha is the Laplacian parameter α, v is the image matrix and w is the enhanced image. The fspecial() command generates the unsharp filter following the operation illustrated in Figure 3.5 and then the filter2() command implements the actual spatial filtering to the original image.

After a number of trials during the course of the study, it was noticed that the specific value of α was not really important; therefore 0.5 was chosen to be a middle value (between 0.0 and 1.0) in order to show the effects, or absence thereof, of unsharp masking to the textural properties of the images. The application of unsharp masking did not produce any noticeable effects to the GLCM texture properties of the images used in this study. The images however appeared to be much clearer to the human eye and perhaps to any pathologist. This can be explained by the fact that unsharp masking only deals with the edges in an image and therefore does not alter significantly the surfaces with important textural information. The usefulness of this image enhancing technique might be more appreciated when morphological analysis is undertaken later in the study.

3.4 Haralick Textural Features

A digital image can be represented as a matrix, or set of matrices, wherein each element contains numerical information about each pixel of the image. Texture can be defined as the mutual relationship among intensity values of neighbouring pixels repeated over an area larger than the size of the relationship (Kulkarni, 2001). Haralick et al. (1973) proposed textural features based on grey-level co-occurrence matrices or GLCMs. These features have been shown to be effective in discriminating microscopic images of colon cancer tissues and cells.

For an N_x x N_y image, with each pixel quantised to N_g levels, let L_x be the horizontal spatial domain, L_y the vertical spatial domain, and G the set of quantised grey levels, such that $L_x = \{1,2,.....N_x\}$, $L_y = \{1,2,.....N_y\}$, and $G = \{1,2,.....N_g\}$. The elements of a GLCM are then the relative frequencies, $P_{ij,}$ with which two neighbouring pixels separated by distance *d* and angle Θ occur on the image, one with grey level *i* and the

other with grey level *j*. For angles quantized to intervals of 45°, then Haralick et al. (1992) defined the un-normalized frequencies as:

P(*i*, *j*, *d*, 0°) = #{[(*k*, *l*) , (*m*, *n*) | *k* – *m* = 0, | *l*-*n* | = *d*, *I*(*k*, *l*) = *i*, *I*(*m*, *n*) = *j*}

P(*i*, *j*, *d*, 45°) = #{[(*k*, *l*) , (*m*, *n*) | (*k* – *m* = *d*, *l*-*n* = - *d*), (*k* – *m* = - *d*, *l* - *n* = *d*), *I*(*k*, *l*) = *i*, *I*(*m*, *n*) = *j*}

P(*i*, *j*, *d*, 90°) = #{[(*k*, *l*) , (*m*, *n*) | |*k* – *m*|= *d*, *l*-*n* = 0, *I*(*k*, *l*) = *i*, *I*(*m*, *n*) = *j*}

P(*i*, *j*, *d*, 135°) = #{[(*k*, *l*) , (*m*, *n*) | (*k* – *m* = *d*, *l*-*n* = *d*), (*k* – *m* = - *d*, *l* - *n* = - *d*), *I*(*k*, *l*) = *i*, *I*(*m*, *n*) = *j*}

where # denotes the number of elements in the set. The co-occurrence matrix can be normalised by dividing each entry by the total number of pairs. Haralick et al. (1973) introduced 14 textural properties derivable from the GLCM. Below is the list of the textural properties used in this study with the 15th property added from the Correlation property to detect hyperchromasia in an image:

Notation:

$p(i, j)$ (i, j) th entry in a normalised gray-tone spatial-dependence matrix, = P(i,j)/R.

$p_x(i)$ ith entry in the marginal-probability matrix obtained by summing the rows of $p(i, j)$, $= \sum_{j=1}^{N_g} p(i, j)$.

N_g Number of distinct grey levels in the quantized image.

$\sum_i$ and $\sum_j$ $\sum_{i=1}^{N_g}$ and $\sum_{j=1}^{N_g}$, respectively.

$$p_y(j) = \sum_{i=1}^{N_g} p(i, j).$$

$$p_{x+y}(k) = \sum_{\substack{i=1 \\ i+j=k}}^{N_g} \sum_{j=1}^{N_g} . p(i,j)$$

where k = 2, 3,…, $2N_g$

$$p_{x-y}(k) = \sum_{\substack{i=1 \\ |i-j|=k}}^{N_g} \sum_{j=1}^{N_g} p(i,j)$$

where k = 0, 1, ….., N_g-1.

1) Angular Second Moment (ASM) or Energy: a measure of homogeneity

$$f_1 = \sum_i \sum_j \{p(i,j)\}^2 \qquad \textbf{Equation 3.2}$$

2) Contrast (inertia): the amount of local variations present

$$f_2 = \sum_{n=0}^{N_{g-1}} n^2 \left\{ \sum_{\substack{i=1 \\ |i-j|=n}}^{N_g} \sum_{j=1}^{N_g} p(i,j) \right\} \qquad \textbf{Equation 3.3}$$

3) Correlation: calculates the linearity of grey level dependencies

$$f_3 = \frac{\sum_i \sum_j (ij) p(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y} \qquad \textbf{Equation 3.4}$$

where μ_x, μ_y, σ_x, and σ_y are the means and standard deviations of p_x and p_y.

4) Sum of Squares: Variance

$$f_4 = \sum_i \sum_j (1-\mu)^2 p(i,j) \qquad \textbf{Equation 3.5}$$

5) Inverse Difference Moment: a measure of local homogeneity

$$f_5 = \sum_i \sum_j \frac{1}{1+(i-j)^2} p(i,j) \qquad \textbf{Equation 3.6}$$

6) Sum Average:

$$f_6 = \sum_{i=2}^{2N_g} i p_{x+y}(i) \qquad \textbf{Equation 3.7}$$

7) Sum Variance:

$$f_7 = \sum_{i=2}^{2N_g} (1 - f_8)^2 \, p_{x+y}(i) \qquad \textbf{Equation 3.8}$$

8) Sum Entropy:

$$f_8 = -\sum_{i=2}^{2N_g} p_{x+y}(i) \log\{p_{x+y}(i)\} \qquad \textbf{Equation 3.9}$$

9) Entropy: characterises texture non-uniformity

$$f_9 = -\sum_i \sum_j p(i,j) \log\{p(i)\} \qquad \textbf{Equation 3.10}$$

10) Difference Variance:

$$f_{10} = \textbf{variance of } p_{x-y} \qquad \textbf{Equation 3.11}$$

11) Difference Entropy:

$$f_{11} = -\sum_{i=0}^{N_{g-1}} p_{x-y}(i) \log\{p_{x-y}(i)\} \qquad \textbf{Equation 3.12}$$

12) , 13) Information Measures of Correlation: the additional properties not included in f_3.

$$f_{12} = \frac{HXY - HXY1}{\max\{HX, HY\}} \qquad \textbf{Equation 3.13}$$

$$f_{13} = (1 - \exp[-2.0(HXY2 - HXY)])^{1/2} \qquad \textbf{Equation 3.14}$$

$$HXY = -\sum_i \sum_j p(i,j) \log(p(i,j))$$

$$HXY1 = -\sum_i \sum_j p(i,j) \log\{p_x(i) p_y(j)\}$$

$$HXY2 = -\sum_i \sum_j p_x(i) p_y(j) \log\{p_x(i) p_y(j)\}$$

where HX and HY are entropies of p_x and p_y

14.) Maximal Correlation Coefficient:

$$f_{14} = (\text{second largest eigenvalue of Q})^{1/2} \qquad \textbf{Equation 3.15}$$

where $Q(i,j) = \sum_k \frac{p(i,k)p(j,k)}{p_x(i)p_y(k)}$

15) Mean:

$$\mu_x = \sum_i \sum_j ip(i,j) \quad \mu_y = \sum_i \sum_j jp(i,j) \qquad \textbf{Equation 3.16}$$

The angular second-moment (ASM), the entropy, the sum entropy, the difference entropy, the information measure of correlation and maximal-correlation features are said to be invariant under monotonic grey-tone transformations (Haralick *et al.*, 1973). The calculation of the features is started by loading the image matrix into memory. The usual quantisation of a grey image is 256 grey levels. This is usually followed by a re-quantisation of the image, say down to 32 grey levels, 16 grey levels, etc, whichever is the chosen new quantisation value to speed-up calculation. The new quantisation determines the size of the GLCM, that is, if say for example the new quantisation is 16 grey levels then the GLCM is a 16 x 16 matrix. A GLCM is calculated for each of 4 directions namely 0°, 45°, 90° and 135°. The distance d of the pixels being compared in calculating a GLCM in this study is held at d=1, based on the suggestion by Zucker and Terzopoulos (1980) to maximise the chi-square significance test. Each entry in a GLCM matrix corresponds to the number of times a pixel with a certain value co-occurs with another pixel value for a given distance d and specified direction. In order to cover all directions surrounding a pixel, each of the co-occurrences of the different pixel values in a GLCM is counted in both directions in order to produce symmetric GLCMs. This has the effect of producing GLCMs for 180°, 225°, 270° and 315° combined with 0°, 45°, 90° and 135° directions, respectively. A GLCM is usually normalised by dividing

each element by R, which is the sum of all the elements in that particular matrix. The GLCMs for the 4 directions are called 'directional GLCMs'. A non-directional GLCM can be produced if the corresponding elements of the directional GLCMs are averaged. This particular step was adopted in this study in order to incorporate image rotation invariance. The end result of all the steps mentioned is a non-directional GLCM with *p(i,j)* as matrix elements; *i* and *j* as row and column indices, respectively. Subsequently, calculation of the 15 features outlined in this section is a straightforward process with the help of a computer.

3.5 Scatter Matrices and Boland *et al.* (1998) Variance Ratio

In Multiple Discriminant Analysis or MDA, a transformation matrix **W** is sought that "in some sense maximises the ratio of between-class scatter to the within-class scatter" (Duda *et al.*, 2001). Based on this measure, the criterion function can be expressed as:

$$\boldsymbol{J(W)} = \frac{|\widetilde{\boldsymbol{S_B}}|}{|\widetilde{\boldsymbol{S_W}}|} = \frac{|\boldsymbol{W^t S_B W}|}{|\boldsymbol{W^t S_W W}|} \qquad \textbf{Equation 3.17}$$

with $\boldsymbol{S_B} = \sum_{i=1}^{c} \boldsymbol{n_i}\,(\boldsymbol{m_i} - \boldsymbol{m})(\boldsymbol{m_i} - \boldsymbol{m})^t$ **Equation 3.18**

$\boldsymbol{S_W} = \sum_{i=1}^{c} \boldsymbol{S_i}$ **Equation 3.19**

$\boldsymbol{S_i} = \sum_{x \in \boldsymbol{D_i}} (\boldsymbol{x} - \boldsymbol{m_i})\,(\boldsymbol{x} - \boldsymbol{m_i})^t$ **Equation 3.20**

$\boldsymbol{m_i} = \frac{1}{\boldsymbol{n_i}} \sum_{x \in \boldsymbol{D_i}} \boldsymbol{x}$ **Equation 3.21**

where $\boldsymbol{S_B}$ = between-class scatter matrix

$\boldsymbol{S_W}$ = within-class scatter matrix

| | = determinant of matrix

c = number of classes

n_i = number of data points in class *i*

D_i = designation of subset or class *i*

x = data point vector

$\mathbf{m}_i$ = sample mean

To select features for classification, Boland *et al*. (1998) used a modified version of the MDA criterion wherein, instead of using scatter matrices, appropriate variance quantities are employed to account for the between-class scatter and within-class scatter. This parameter or criterion is sometimes referred to as "variance ratio" in this thesis to give it a simpler name. Mathematically, this criterion can be expressed as a ratio with the between-class variance as numerator and the sum of the within-class variances as denominator (see equation 3.22). Features with high variance ratios are considered to exhibit a good discriminating characteristic.

$$\boldsymbol{variance\ ratio} = \frac{\boldsymbol{var(f)}}{\sum_{\boldsymbol{c}} \boldsymbol{var(f_c)}} \qquad \textbf{Equation 3.22}$$

where $\boldsymbol{f}_c$ ➔ feature values from class c

$\boldsymbol{f}$ ➔ feature values from all classes

$var(\,)$ ➔ variance operator

As an illustrative example on how to calculate variance ratios given data points characterised by different property values, assume that there are 5 subjects, each described by 4 features and has 2 classes: A and B. Table 3.2 presents this data in tabular form. For simplicity, assume that all the features have a common scale such that normalisation of data is not necessary. From the given table in this illustrative example, it can be observed that subject1 and subject2 both belong to class A, while subject3, subject4 and subject5 belong to class B. This table can be conveniently expressed as a matrix. Since a variance ratio is computed for a particular feature or property, then in this example, there should be 4 variance ratios which will be compared later on since there are 4 features given.

Table 3.2 Illustrative example on how to calculate the variance ratios used by Boland *et al.* (1998)

	Feature1	Feature2	Feature3	Feature4	Class
Subject1	20	50	100	5	A
Subject2	18	40	150	5	A
Subject3	10	55	200	4	B
Subject4	5	52	205	5	B
Subject5	2	45	125	3	B

Using Equation 3.22 to calculate the variance ratio for Feature1, three variances are computed from Table 3.2. The first variance is the variance in Class A:

$$\text{variance of } \{20, 18\} = 2.000$$

The second variance is the variance in Class B:

$$\text{variance of } \{10, 5, 2\} = 16.333$$

The third variance is the variance in the entire data set (but limited only to Feature1):

$$\text{variance of } \{20, 18, 10, 5, 2\} = 62.000$$

Therefore the variance ratio for Feature1 can now be calculated as:

$$\text{variance ratio for Feature1, } VR1 = \frac{\text{variance of } \{20,18,10,5,2\}}{\text{variance of } \{20,18\} + \text{variance of } \{10,5,2\}} = 3.382$$

The variance ratios for the other features are calculated in the same manner. Table 3.3 and Figure 3.6 summarise the calculated variance ratios of all the features using MS Excel and Equation 3.22. The rule is: the higher the variance ratio, the better the feature in discriminating between classes. Therefore from Figure 3.6, it is clear that Feature1 stands out; it is a good feature to use to classify the subjects into Class A or Class B, compared to the other features. One disadvantage in using this variance ratio to select features is the fact that it does not consider the combined effects of features. There might be hidden relationships among different feature spaces that could be explored. However in this study, the use of the variance ratios proved to be very effective as exhibited by the good classification rates using ANFIS classifiers.

Table 3.3 The calculated variance ratios in the give illustrative example in Table 3.2

	Feature1	**Feature2**	**Feature3**	**Feature4**
Variance Ratio, VR	3.382	0.462	0.650	0.800

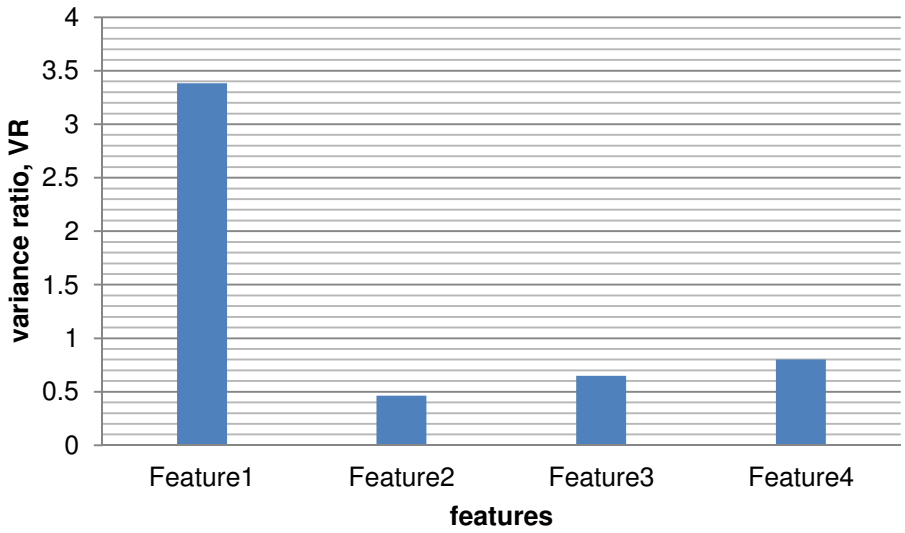


Figure 3.6 Equivalent plot of the variance ratios in Table 3.3 for the given illustrative example.

3.6 Kohonen Self-Organising Map (KSOM)

Kohonen self-organising map or KSOM is an unsupervised artificial neural network architecture that was popularised by Teuvo Kohonen. It is also known as Kohonen self-organising network or KSON. The operating principle of this network is based on the characteristic of the animal brain to organise spatially the internal representations of information (Kohonen, 1990). Shown in Figure 3.7 is a schematic representation of the Kohonen self-organising network.

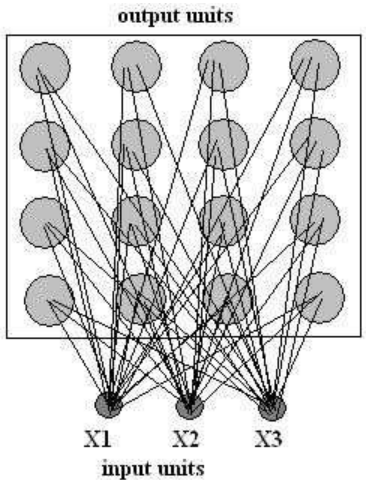


Figure 3.7 Schematic representation of the Kohonen self-organising network or KSON

The output units, also known as neurons or nodes, and the input units are fully connected. The input units receive the properties of the data under classification. This architecture allows the process known as competitive learning to take place. The neurons 'compete' among themselves and the winner and its neighbours are rewarded by allowing their weights to be updated. The winning neuron is chosen by selecting the node that is closest to the input data based on Euclidian distance. Let $\mathbf{w}_{ij}$ denote the neuron weight vector and **x** the input pattern. The winning neuron is selected using

$$\min_{ij} \left\| x - w_{ij} \right\|.$$

The winning neuron is usually called the best matching unit or BMU. The weight vectors of the winning neurons are updated according to:

$$w_{ij}(t+1) = \begin{cases} w_{ij}(t) + \alpha(t)[x - w_{ij}(t)], if\ (i,j) \in N_c(t) \\ w_{ij}(t), if\ (i,j) \notin N_c(t) \end{cases}$$ **Equation 3.23**

where $\alpha(t)$ represents the adaptive learning rate and $N_c(t)$ is the neighbourhood of the winning neuron at iteration t. Both $\alpha(t)$ and $N_c(t)$ are decreased at every iteration according to some decreasing function. The entire process of learning, which is characterised by the updating of the weight vector, generates a topographic mapping of the input to the output and results in a reduction in the dimension of the input space (Karray and de Silva, 2004).

The quantisation error is a widely used measure among many that have been used to evaluate the quality of a self-organising map (Uriarte and Martin, 2005). It is defined as the average distance from sample vectors to its best matching unit or BMU (Kiviluoto, 1996):

$$\boldsymbol{qe} = \frac{\mathbf{1}}{\boldsymbol{N}} \sum \left\| \bar{\boldsymbol{x}}_{\boldsymbol{i}} - \boldsymbol{m}_{\bar{\boldsymbol{x}}_{\boldsymbol{i}}} \right\|$$ **equation 3.24**

where *qe* is the quantisation error, *N* is the number of data vectors, and $m_{\bar{x}_i}$ is the best matching prototype of the corresponding $\bar{x}_i$ data vector. The optimal map is expected to have a minimal average quantisation error.

3.7 Genetic Algorithms (GA)

Genetic algorithms or GA were first proposed by Holland (1975). Genetic algorithms belong to a class of adaptive search population-based techniques that can

be applied to solve optimisation problems without using derivatives or gradients. The general term that is used to refer to this class of population-based search techniques is evolutionary computing. Unlike single-point-based optimisation algorithms, evolutionary computing methods operate on populations of candidate solutions to find the global minimum or maximum. Single-point-based search techniques are susceptible to getting stuck at local maxima and therefore fail completely to find the global maximum. Evolutionary computing techniques do not have this problem since the entire search space is strategically covered by the search procedure through the use of population of solutions and genetic mutation operator. The basic principle in evolutionary algorithm is to emulate the natural selection process. Each solution candidate is represented as an array of strings to form a chromosome of that particular individual. During the implementation of the search process, evolutionary operators such as *selection*, *recombination* or *crossover*, and *mutations* are applied to the different individuals or *chromosomes*. Individuals are evaluated based on a fitness function that gives the 'fitness' of candidate solutions and entire populations as well. Cordon *et al*. (2001) outlined the following issues that must be addressed in order to implement GA:

- genetic representation of candidate solutions,
- creation of initial population of solutions,
- choice of fitness function or evaluation function of each individual,
- genetic operators to produce new variants during recombination, and
- values of GA parameters e.g., population size, number of generations, probabilities in the application of genetic operators [selection probability, crossover probability, mutation probability].

The genetic representation of candidate solutions can be either binary or real numbers. In GA, each candidate solution is identified as a *chromosome*, which is composed of a contiguous arrangement of bits or numbers, each called a *gene* and

usually handled in computer memory as a 1-dimensional array e.g., row array. The *allele* of a gene is the value of that particular gene. The *phenotype* refers to the physical makeup, while *genotype* refers to a specific combination of genes of a candidate solution or organism. Figure 3.8 outlines the steps taken in implementing a typical GA.

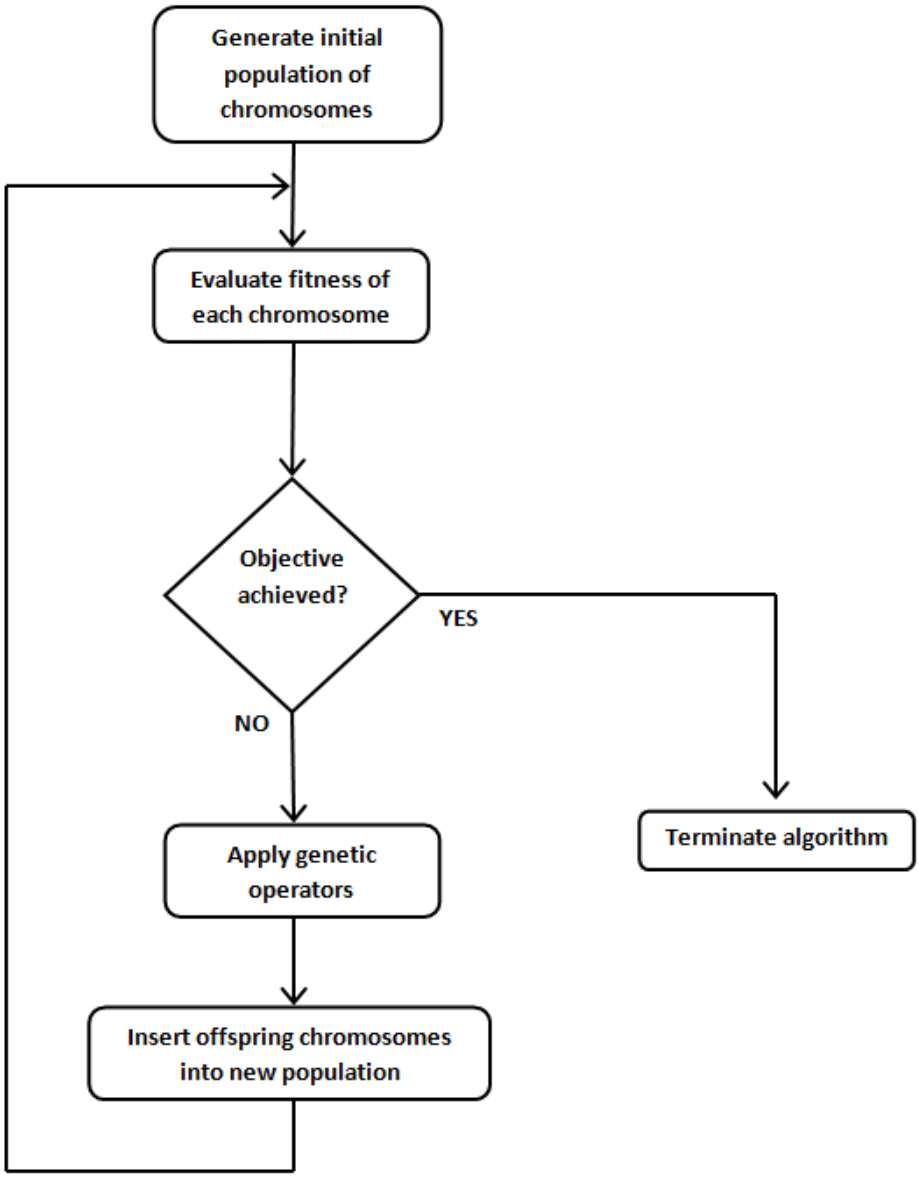


Figure 3.8 Schematic flowchart of a genetic algorithm

The algorithm usually starts by generating the initial population of individuals, each of which is encoded in a chromosome. The number of populations, N, is normally held fixed throughout the algorithm and must be known a-priori. As mentioned previously, encoding can be performed in binary or real numbers. GA implementations with chromosomes that are encoded with real numbers as genes are known as Real-Coded Genetic Algorithms or RCGA. The initial population can be generated in a random fashion or according to some more elaborate scheme. Immediately upon assembling the initial population, the individual chromosomes are evaluated though their fitness values using a fitness function that has been set as part of the parameters of the GA itself. In almost all cases, the initial population is allowed to undergo genetic evolutionary transformation to explore the possibilities of producing 'good' candidate solutions. This process is repeated as illustrated in Figure 3.8 until a termination condition is met. Throughout the entire process, it is advisable to monitor the output parameters, such as fitness values, in order to analyse the quality of the generated individuals and monitor the convergence of solutions to the global maximum or minimum, that is, if it exists.

Attempts to combine or integrate GA and KSOM have been mainly aimed at improving KSOM. Polani and Uthmann (1992) used GA to improve a Kohonen net topology by using a transcription rule to represent the net topology by genotype. The Kohonen net was trained and then subjected to map quality test which served as the fitness function. The fitness function or quality test used was essentially a measure of the average distance from an input vector to the vertex it activates – a smaller distance yields a higher quality function which means a better adaptation to the input space. In another study, Huang and Hung (1995) proposed to use GA in order to improve the initialisation of the KSOM. The fitness function chosen was the error vector between training set vectors and their nearest weight vectors. One of the aims in this study is to propose an algorithm that incorporates KSOM and GA together specifically to tackle

feature selection. The motivation to use GA with KSOM comes from the unsupervised nature of KSOM and from the very efficient search method of GA. The use of KSOM allowed for the investigation of tendencies of 'similar' data points to cluster together without relying on the classes given by human experts. Usually when one needs to select a set of features, a classifier is used to evaluate the accuracy for the chosen feature set. This approach presupposes that there is no question as to which classifier must be used. In this study, choosing a classifier is part of the investigation; therefore another approach is necessary. The use of KSOM avoids this problem since the classes of the output (training) data are not used. The classification part is achieved by clustering through the self-organisation in a Kohonen map. The GA comes into the picture as part of the fitness function to be able to evaluate map quality.

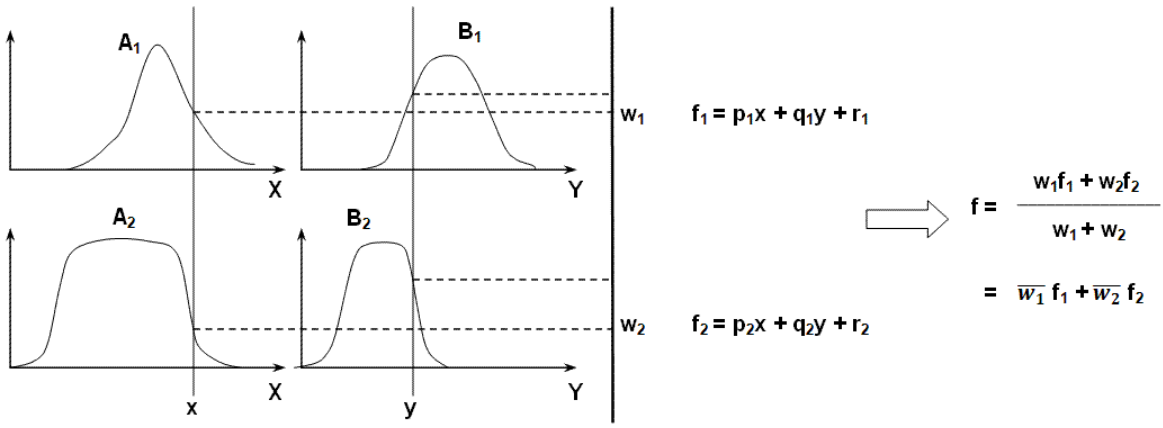
3.8 Adaptive-Network-Based Fuzzy Inference System (ANFIS)

ANFIS stands for *Adaptive-Network-based Fuzzy Inference System* or semantically equivalently *Adaptive Neuro-Fuzzy Inference System*. It is a hybrid neuro-fuzzy system proposed by J-S Jang (1993). It is well-known that Fuzzy Inference Systems or FIS are very useful because they allow us to put linguistic information from human experts into computer algorithms. However, a main drawback is the lack of facility to automatically learn from data, which, incidentally is the strength of feed-forward artificial neural networks or ANN. ANFIS combines the advantages of FIS and ANN into a single implementation by designing a feed-forward ANN that performs the operations in the FIS. The ANN training method has also been improved in ANFIS by a hybrid learning scheme. ANFIS uses only the Sugeno-type of fuzzy system with the following constraints (Karray and de Silva, 2004):

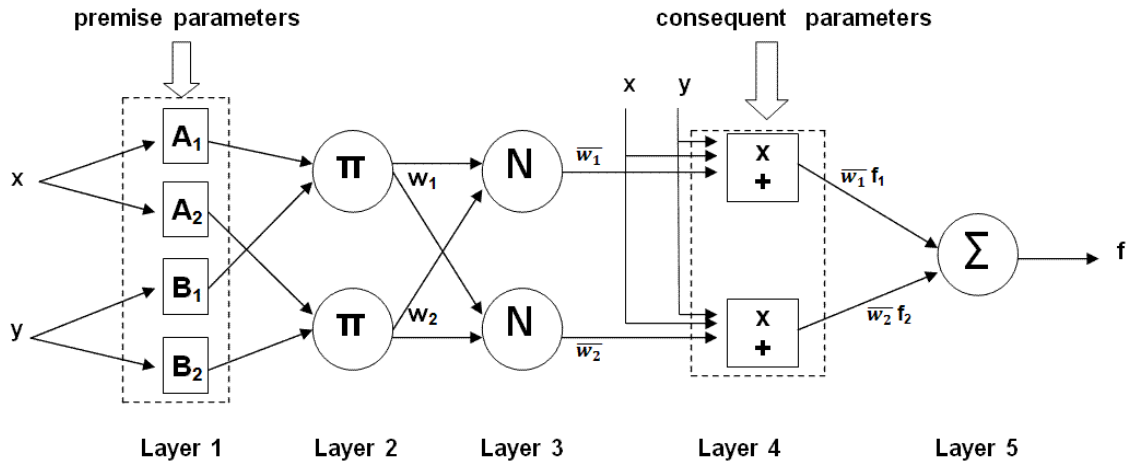
- Zero or 1st order Sugeno-type systems

- A single output obtained using a weighted average defuzzification method
- The weight of each rule is unity.

Figures 3.9a and 3.9b show an example of a 2-input first order Sugeno fuzzy model with two rules and an equivalent ANFIS architecture.



(a)



(b)

Figure 3.9 (a) and (b) 2-input 1st order Sugeno fuzzy model with 2 rules and the equivalent ANFIS architecture based on Fig. 28 of Jang and Sun (1995)

Layer 1 implements fuzzification of crisp input data considering the premise parameters such as membership function parameters. Layer 2 determines the firing strength of a rule by applying T-norm operators on the fuzzy values. Layer 3 normalises

the firing strengths produced by Layer 2 while Layer 4 calculates the input for Layer 5 by using the normalised firing strengths and the consequent parameters. Finally, Layer 5 computes the overall output but adding together the outputs of Layer 4. ANFIS uses a hybrid learning algorithm wherein the forward pass employs least-squares estimate (LSE) to identify the consequent parameters while the backward pass uses gradient descent to update the premise parameters. Table 3.4 summarises the activities in the ANFIS hybrid learning procedure. In the forward pass, the premise parameters are held fixed while the consequent parameters are calculated using least-square estimate or LSE. In the backward pass, which is analogous to the back-propagation in the standard ANN, the consequent parameters are held fixed while the premise parameters are calculated using gradient descent.

Table 3.4 Activities in each pass in the ANFIS hybrid learning procedure

	Forward Pass	**Backward Pass**
Premise Parameters	Fixed	Gradient Descent
Consequent Parameters	Least-Squares Estimate (LSE)	Fixed
Signals	Node Outputs	Error Signals

There are numerous artificial intelligence algorithms and methodologies that can be used to solve the classification problem in this study. However, ANFIS was chosen for specific reasons. First and foremost is the fact that training data is available. This means that a supervised classifier can be used. Among conventional supervised classifiers, BPNN or back-propagation neural networks are more preferred since they

work in parallel and are adaptive (Kulkarni, 2001). In addition, BPNN are said to provide a greater degree of robustness or fault tolerance. These excellent qualities are enhanced by incorporating a fuzzy inference system, or FIS, which allows the processing or production of linguistic information. The main drawback of using BPNN is that all operations take place within a black box. It is impossible to make sense of the 'logic' or knowledge contained in a trained neural network. This drawback is eliminated by incorporating FIS into a BPNN. The combination of the two allows the production or extraction of knowledge from a set of numerical data. There can be a number of ways that BPNN and FIS can be integrated, but ANFIS has been strategically chosen because it has proven to be very useful in various research studies. In addition, it uses a first-order Sugeno fuzzy model which is very much applicable to the nature of the output in this research. One of the novel ideas in this research is to express the classification output as a spectrum of values from 0.0 to 1.0 to characterise the varying degrees of abnormality in a colonic image, instead of simply saying that a sample is normal, dysplastic or cancerous. This approach assumes that there is a linear relationship between the properties and the output variable. This is precisely the premise behind the use of the first-order Sugeno FIS model. The assumption of linearity between the input and output variables is the reason in adopting the first-order Sugeno FIS model. Along with the advantage of having a hybrid training scheme, which is more advantageous than the pure back-propagation algorithm in a BPNN, ANFIS was therefore the classifier of choice in this research.

3.9 Confusion Matrix

A confusion matrix is a table of numbers arranged in a square matrix showing the number of correct classifications and number of misclassifications of a classifier. In this study, the columns represent the expected classifications while the rows represent the classifications made by the classifier being evaluated. Ideally, the elements in the main diagonal must be equal to the number of samples in each particular class while the off-diagonal elements must all be zero. In other words, the ideal confusion matrix is a diagonal matrix. The sum of the diagonal elements in a confusion matrix when normalised gives the percent accuracy of the classifier. Table 3.5 shows an example of a confusion matrix with 3 classes.

Table 3.5 Example of a confusion matrix with 3 classes. The columns are the expected classifications while the rows are the predicted classes of the classifier.

	Expected Class A	Expected Class B	Expected Class C
Predicted Class A	a	b	c
Predicted Class B	d	e	f
Predicted Class C	g	h	i

A widely used single-value information that can be extracted from a confusion matrix is known as accuracy of the classifier in question. This is calculated by adding elements a, e and i in Table 3.5.

Illustrative example:

Consider the confusion matrix:

$$CM = \begin{bmatrix} 65 & 5 & 0 \\ 5 & 55 & 6 \\ 0 & 10 & 64 \end{bmatrix}$$

where the 3 classes LOW, MEDIUM and HIGH, are assigned from left to right and from top to bottom in the given matrix. With reference to Table 3.5, the left-most column means that out of 70 expected LOW classifications, 5 cases were misclassified as MEDIUM and none was misclassified as HIGH. The middle column can be interpreted as 55 correct classifications as MEDIUM, while 5 cases were misclassified as LOW and 10 cases were misclassified as HIGH; still a total of 70 cases for MEDIUM. Finally in the right-most column, out of 70 cases, 64 HIGH cases were correctly classified while 6 cases were misclassified as MEDIUM and none was misclassified as LOW. Using the elements in the main diagonal {65, 55, 64}, the sum 184 accounts for all the correct classifications. The off-diagonal elements are the misclassifications. By first normalising the entries in the given matrix, the percent accuracy can be calculated to be 87.62%.

3.10 Software and Hardware Used

The software development platform that was used in the most part of the algorithms in this study was MATLAB version R2009a by Mathworks. The hardware used was an Acer desktop computer with the following specifications:

Processor :Intel(R) Core(TM)2 Duo CPU E8400 @3.00 GHz 3.00 GHz

Memory (RAM) :2.00 GB

System Type :32-bit Operating System

Chapter 4 – TEXTURAL FEATURE CALCULATION AND FEATURE SELECTION

As far as the image features are concerned, the scope of this study was limited to the use of all of the 14 textural features introduced by Haralick *et al.* in 1973. The 'sub-feature' known as the *mean* has also been included in the list of possible features, making the total equal to 15 in an attempt to account for the darkening of pixels due to hyperchromasia specifically for non-normal images. Each of these features was calculated from a non-directional Grey Level Co-Occurrence Matrix or GLCM produced by averaging element-by-element the directional GLCMs at directions 0°, 45°, 90°, and 135°. The distance used in calculating all the GLCMs was 1 pixel, based on the suggestion by Zucker and Terzopoulus (1980) to optimise GLCM by maximising chi-square significance test. In identifying the sets of discriminating features in this study, two processes were compared: selection based on the Boland *et al*. (1998) variance ratio and selection using genetic algorithm and Kohonen self-organising map.

4.1 Production of Digital Images from Microscopic Slides

The images used in this study were derived from slides and cases randomly chosen from the 2007 and 2008 surgical pathology files of Medical Center Manila Hospital, previously diagnosed as colonic adenocarcinoma, adenomatous polyps from the colon, as well as resection planes of the colonic resections without tumour to serve as controls. These slides were routinely processed using a Sakura tissue processor and cut at 8 micra using a standard microtome. All were stained with hematoxylin and

eosin. All images were taken at 400x magnification using an Olympus DP20 digital photomicrography apparatus mounted on an Olympus microscope (trinocular) at 1200x1800 dpi resolution. Figure 4.1 shows a diagrammatic picture of the imaging setup used to produce the digital images in this study.

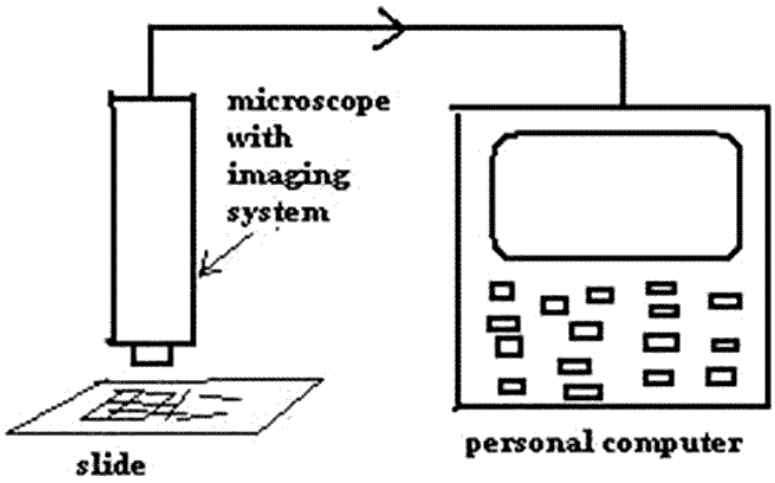


Figure 4.1 Schematic diagram of the imaging system

There were a total of 300 1200x1600-pixel images produced for this study. Immediately after the images were received from the pathologist, it was observed that the 1200x1600-pixel-size of each image was unnecessarily large. For purposes of classification performance comparison between human pathologists and the artificial classifier systems developed here, each image was resized down to 300x400 pixels and converted to a monochromatic image. This size seemed perfect for printing on an A4 sheet of bond paper with 3 monochromatic images per sheet with each image having dimensions of approximately $3\,{}^{1}/_{8}$ inches by $4\,{}^{3}/_{16}$ inches. This is how the images in the pathologist survey form are arranged. An added benefit of dealing with a smaller image size is faster computer processing. The size of the printed images turned out to be quite acceptable to the pathologists who took part in the survey carried out in relation to this study. The aim of the survey was basically to establish a

benchmark for the performance of the artificial classifier systems developed in this study.

Three classes or cases were considered, namely, 'normal', 'adenomatous polyp', and 'cancerous' with each class having 100 images. The adenomatous polyp case represented dysplasia or the middle ground between the normal and cancerous cases. The training and testing sets were formed by following a 70:30-ratio. This means that the training set had 210 images while the testing set had 90 images. According to Ye (2003), in dividing samples into training and testing sets, a 2/3 to 1/3 portion is reasonable; and for tens of thousands of samples a smaller percentage for testing might be considered. Before a decision was made regarding how to divide the samples, three percentage choices were considered for the testing set: 20%, 30% and 40%. The 30% choice was selected since it was seen as the 'safer' value being the middle value between 20% (too small) and 40% (too large). Each of the images was randomly selected from each class. To manage all the images, a file naming system was put in place. Before the images were segregated to form the training and testing sets, each image was renamed using a YXXX coding or naming system. The 'XXX' is for the image count from 001 to 100 while the 'Y' is for the image class label: 'n' for normal images, 'p' for adenomatous polyp images, or 'c' for cancerous images.

It was a surprise to learn in this research that the gathering of the images turned out to be more difficult than expected. It was observed that many pathologists in Metro Manila in the Philippines were unwilling to engage in research collaboration and/or did not possess the skill and resources to provide microscopic digital images of good quality for the task in this study. The search for a 'credible' and willing pathologist ended at the College of Medicine of the University of the Philippines where no less than the Pathology Department chair himself agreed to take-up the challenge to produce the images needed. For this reason, he has been specifically mentioned in the Acknowledgement section of this thesis.

4.2 Feature Selection Using Variance Ratio

The inspiration of this feature selection approach comes from Boland *et al*. (1998) where a modified version of the MDA criterion was used. Instead of using usual scatter matrices in MDA, appropriate variance quantities were employed to account for the between-class scatter and within-class scatter. The variance ratio used by Boland *et al*. (1998) is given by equation 3.22 and also reprinted here in this section for easy reference:

$$variance\ ratio = \frac{var(\boldsymbol{f})}{\sum_c var(\boldsymbol{f}_c)} \qquad \text{Equation 3.22}$$

where $\boldsymbol{f}_c$ ➔ feature values from class c

$\boldsymbol{f}$ ➔ feature values from all classes

$var(\,)$ ➔ variance operator

Application of equation 3.22 was limited to the set of training images. Normalisation of data, such as dividing each value by the maximum, is not necessary since the variance ratio in itself is already a form of normalisation.

Before the variance ratios could be calculated, all 15 textural properties from each training image had to be generated. This was achieved by saving the properties of all the training images in a single data file with file extension name 'data'. The convention on writing data in a *.data file allows one to specify the number of variables or properties involved and also the names of the properties themselves at the header section of the file. The values of the properties are written in a row format with blank space and return for newline as dividers of entries. Incidentally, the generated *.data files can also be conveniently opened by a simple text file editor such as Notepad. A program was written to implement the calculation of the 15 prospective properties. The desired quantisation level of the GLCM, the location of the training images, and file name of the *.data file where the data will be written were the necessary inputs to the

program prior to execution. The workhorse of the program was a function that computes the quantised and normalised GLCM before each of the 15 textural properties could be calculated and tabulated in a *.data file. Another function was written to calculate the variance ratios from the *.data file. As mentioned earlier, normalisation of the data was not necessary since the variance ratio in itself is already a form of normalisation. The calculated variance ratios for the different properties for the whole training image set were plotted using MS Excel since it seemed to produce better looking horizontal bar graphs with a mixture of numeric and non-numeric data compared to MATLAB. The following is a summary of series of steps that were undertaken to produce a single horizontal bar graph representing the variance ratios of the prospective textural properties.

Procedure to produce a variance ratio bar graph:

1. Use the script program “image2FeatureDATAFile.m” to calculate the textural properties of each training image and store in a .data file.
2. Calculate the variance ratio of each textural property for the entire training image set using the function program “glf_computeVarianceRatio.m” with the corresponding .data file as input.
3. Using MS Excel, produce the corresponding horizontal bar graph from the computed variance ratios. Unlike MS Excel, MATLAB R2009a does not seem to support strings as values in the vertical axis of a horizontal bar graph. This is why MS Excel was used to produce the bar graphs.

Copies of the MATLAB codes used in this study such as “image2FeatureDATAFile.m” and “glf_computeVarianceRatio.m” can be found in Appendix A.3 and Appendix A.4, respectively.

Although it was already decided to adopt 300x400 pixels as the standard image size in this study based on the survey forms given to (human) pathologists, there were still questions as to the possible effects on the properties of the resize that was made

on each of the image and also the possible effects if histogram equalisation and edge enhancement were to be performed prior to any classification. This curiosity led to the generation of variance ratio bar graphs of monochromatic training images with different quantisation levels and image sizes. The different image sizes investigated were 1200x1600 pixels (the original size), 900x1200 pixels (75% of original size), 600x800 pixels (50% of the original size), and 300x400 pixels (25% of the original size). Bar graphs were also generated for 32, 24 and 8 quantisation levels for the GLCM at 300x400-pixel image size. Additional bar graphs were also generated for all the images where histogram equalisation and unsharp masking were performed prior to the calculation of textural features at 300x400-pixel image size and 16 quantisation levels. Table 4.1 summarises the variance ratios for the different options just mentioned while Figures 4.2 to 4.10 show all the corresponding bar graphs generated.

Table 4.1 Variance ratios of the training images with different sizes, quantization levels, and basic processing*

Textural properties	A	B	C	D	E	F	G	H	I
[1]ASM	0.3347	0.3307	0.3384	0.333	0.3475	0.3443	0.3403	0.3743	0.3318
contrast	0.5042	0.505	0.5011	0.4981	0.4953	0.4988	0.5015	0.3888	0.5037
mean	1.2328	1.2517	1.2403	1.2251	1.2399	1.24	1.24	0.3333	1.2515
variance	0.4585	0.5108	0.4588	0.5036	0.4638	0.4635	0.4627	0.3403	0.5094
correlation	0.415	0.391	0.3998	0.3848	0.3928	0.3912	0.3954	0.3897	0.3893
[2]IDM	0.4113	0.3771	0.442	0.4022	0.4726	0.4669	0.4566	0.3713	0.3868
sumAverage	1.2328	1.2517	1.2403	1.2251	1.2399	1.24	1.24	0.3333	1.2515
sumEntropy	0.4175	0.4331	0.4049	0.3968	0.4181	0.4122	0.4046	0.3385	0.4214
sumVariance	1.0398	1.03	1.0514	1.0147	1.0503	1.0498	1.05	0.3786	1.032
entropy	0.4245	0.4027	0.432	0.4057	0.438	0.4373	0.4341	0.3794	0.4037
differenceVariance	0.5339	0.5317	0.5309	0.5306	0.5018	0.5102	0.5189	0.3892	0.5321
differenceEntropy	0.4872	0.4422	0.4995	0.4564	0.4825	0.4854	0.4866	0.3703	0.4491
[3]IMC12	0.4084	0.375	0.4213	0.3745	0.4569	0.4523	0.4398	0.3842	0.3743
[3]IMC13	0.4279	0.4	0.4328	0.4075	0.4455	0.4528	0.4512	0.3965	0.4016
[4]MCC	0.5686	0.4272	0.4646	0.3816	0.4673	0.467	0.4676	0.4522	0.4045

* A – 300x400 pixels, 32 quantisation levels; B - 300x400 pixels, 24 quantisation levels; C - 300x400 pixels, 16 quantisation levels; D - 300x400 pixels, 8 quantisation levels; E - 1200x1600 pixels, 16 quantisation levels; F - 900x1200 pixels, 16 quantisation levels; G - 600x800 pixels, 16 quantisation levels; H - 300x400 pixels, 16 quantisation levels (histogram equalised); I - 300x400 pixels, 16 quantisation levels (unsharp masking, $\alpha = 0.5$).

[1] Angular second moment

[2] Inverse Difference Moment

[3] Information Measures of Correlation (f_{12} and f_{13})

[4] Maximal Correlation Coefficient

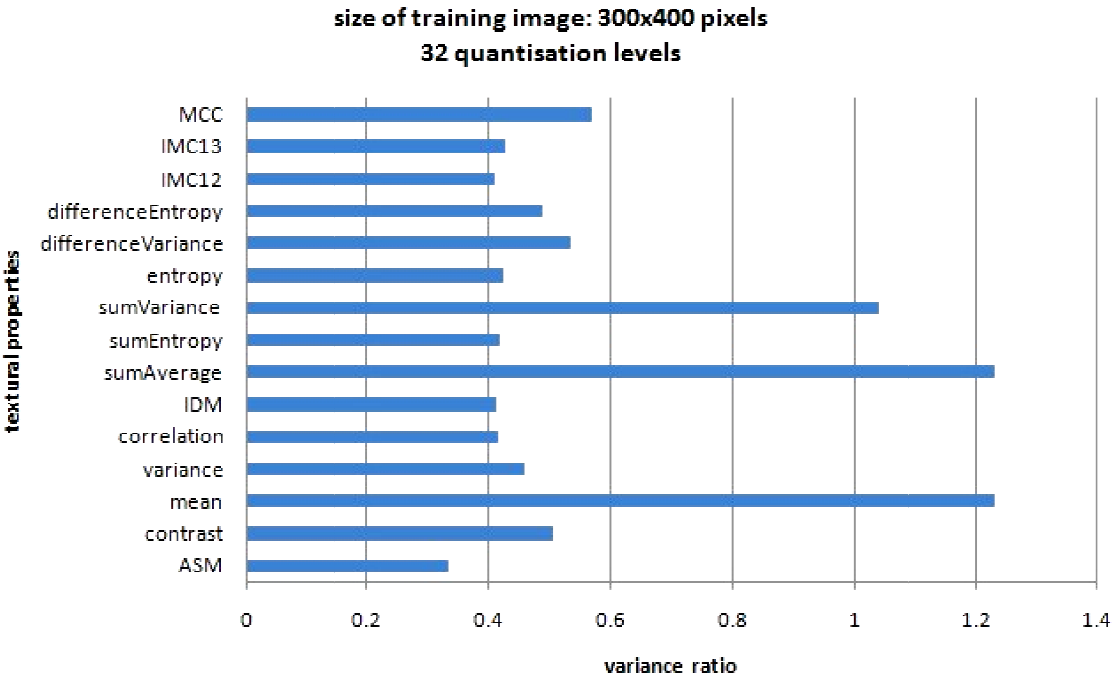


Figure 4.2 Variance ratio bar graph for training images with 300x400 pixels image size and 32 quantisation levels

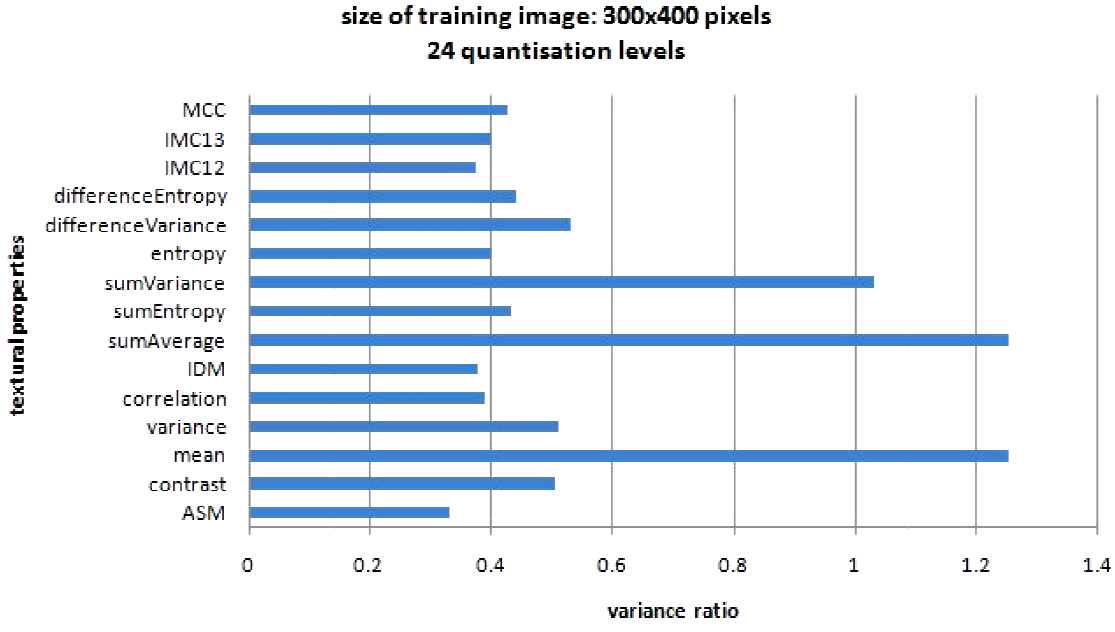


Figure 4.3 Variance ratio bar graph for training images with 300x400 pixels image size and 24 quantisation levels

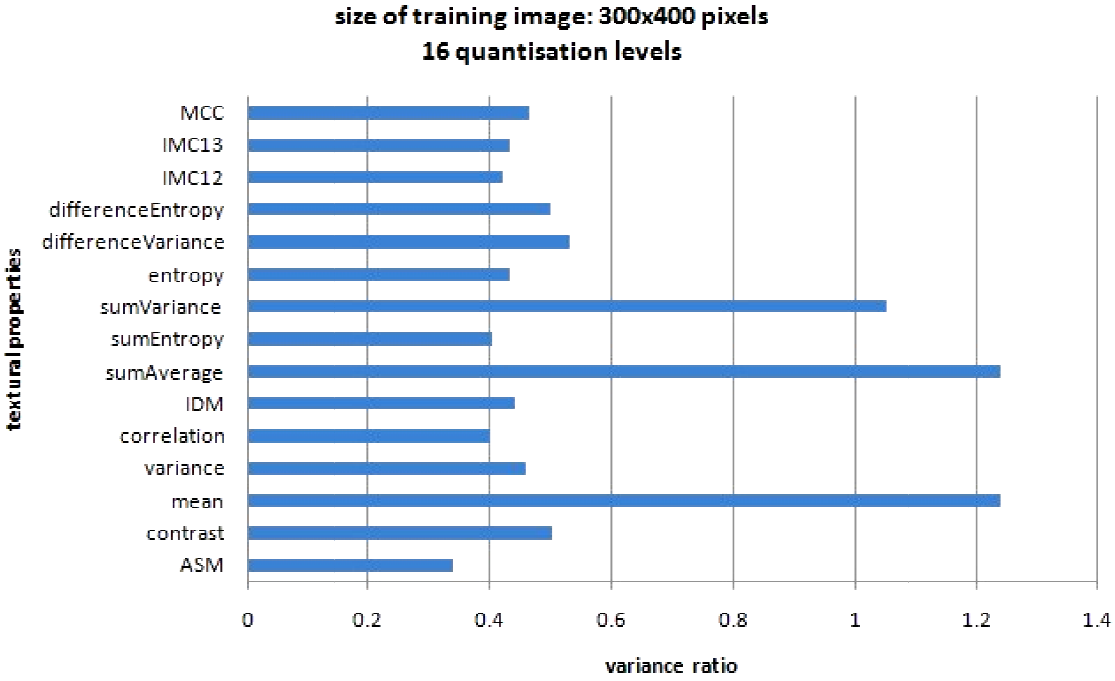


Figure 4.4 Variance ratio bar graph for training images with 300x400 pixels image size and 16 quantisation levels

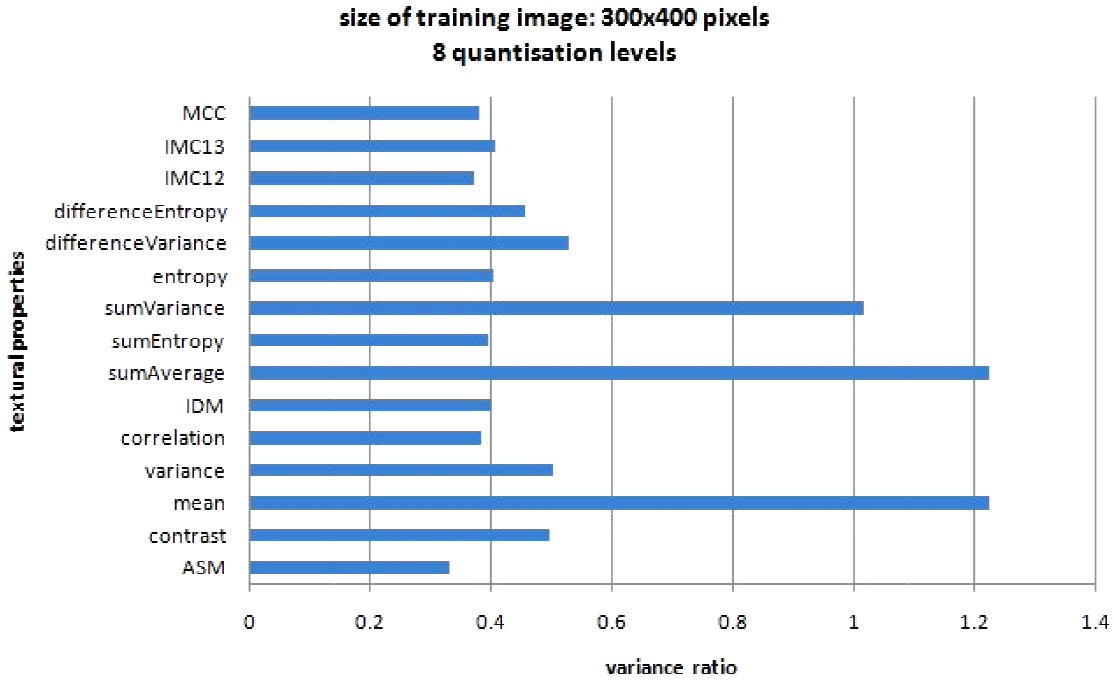


Figure 4.5 Variance ratio bar graph for training images with 300x400 pixels image size and 8 quantisation levels

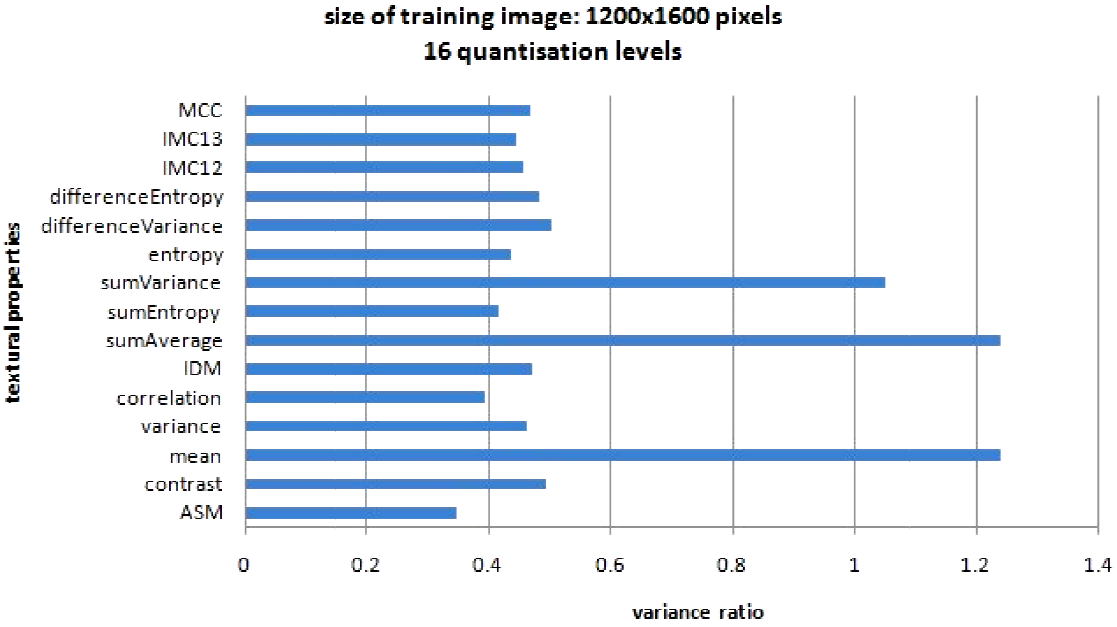


Figure 4.6 Variance ratio bar graph for training images with 1200x1600 pixels image size and 16 quantisation levels

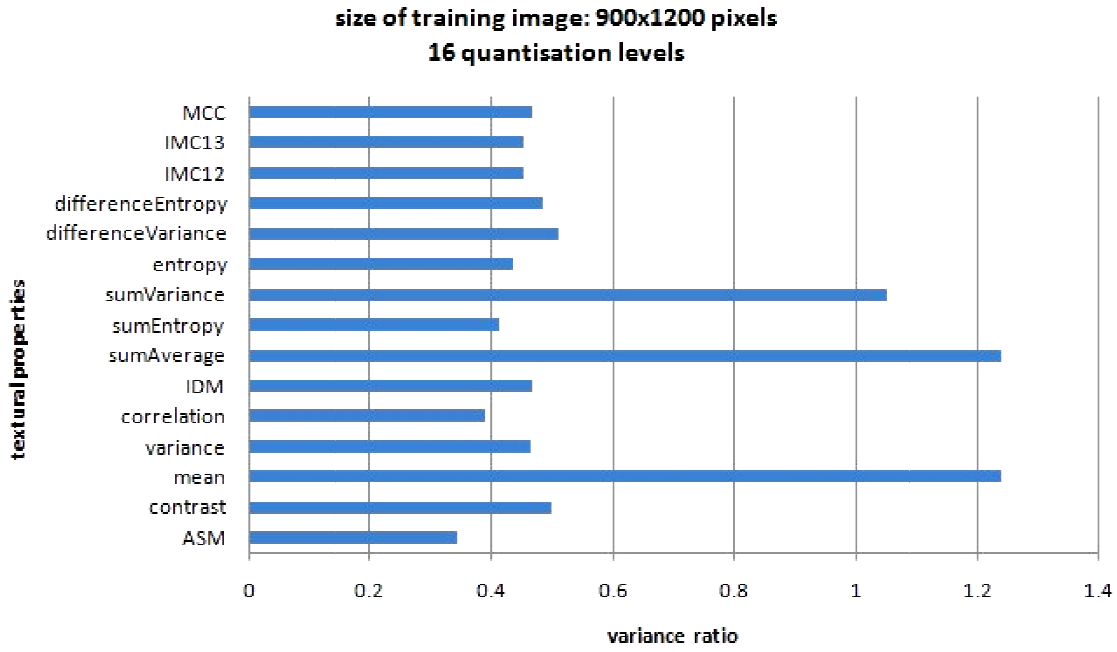


Figure 4.7 Variance ratio bar graph for training images with 900x1200 pixels image size and 16 quantisation levels

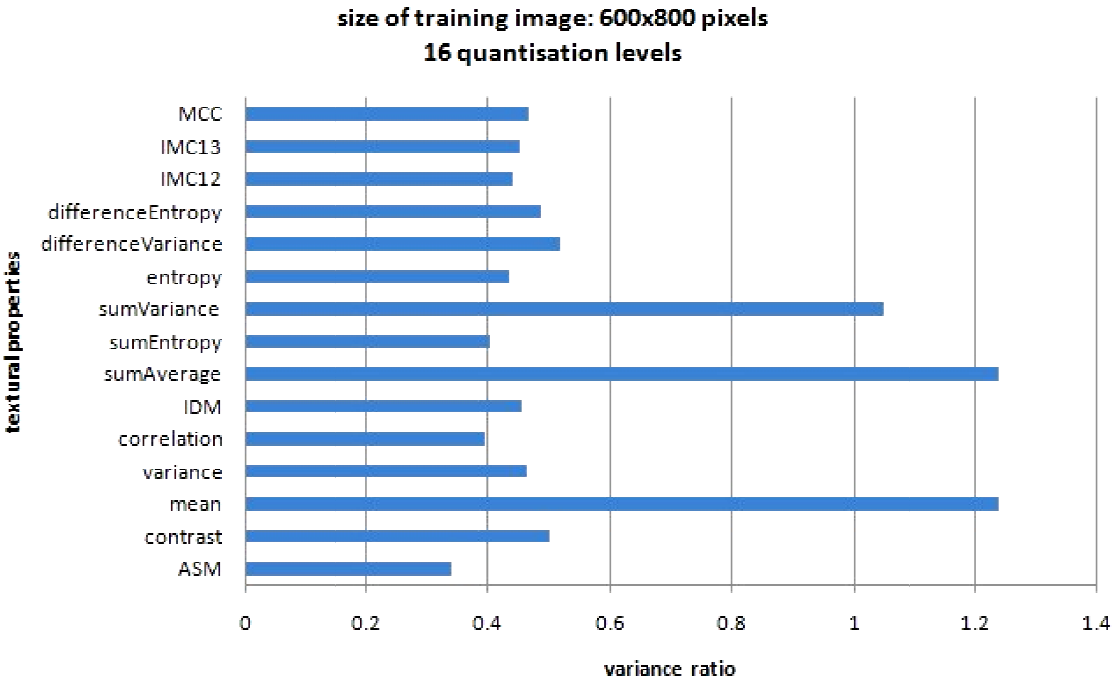


Figure 4.8 Variance ratio bar graph for training images with 600x800 pixels image size and 16 quantisation levels

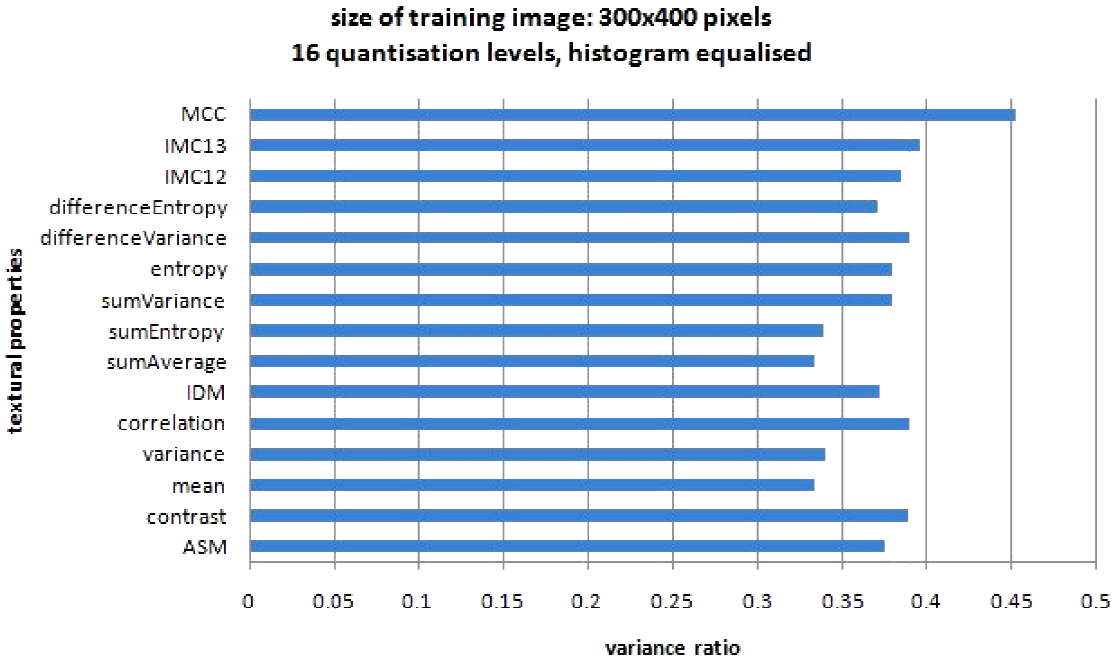


Figure 4.9 Variance ratio bar graph for training images with 300x400 pixels image size, 16 quantisation levels, and histogram equalised

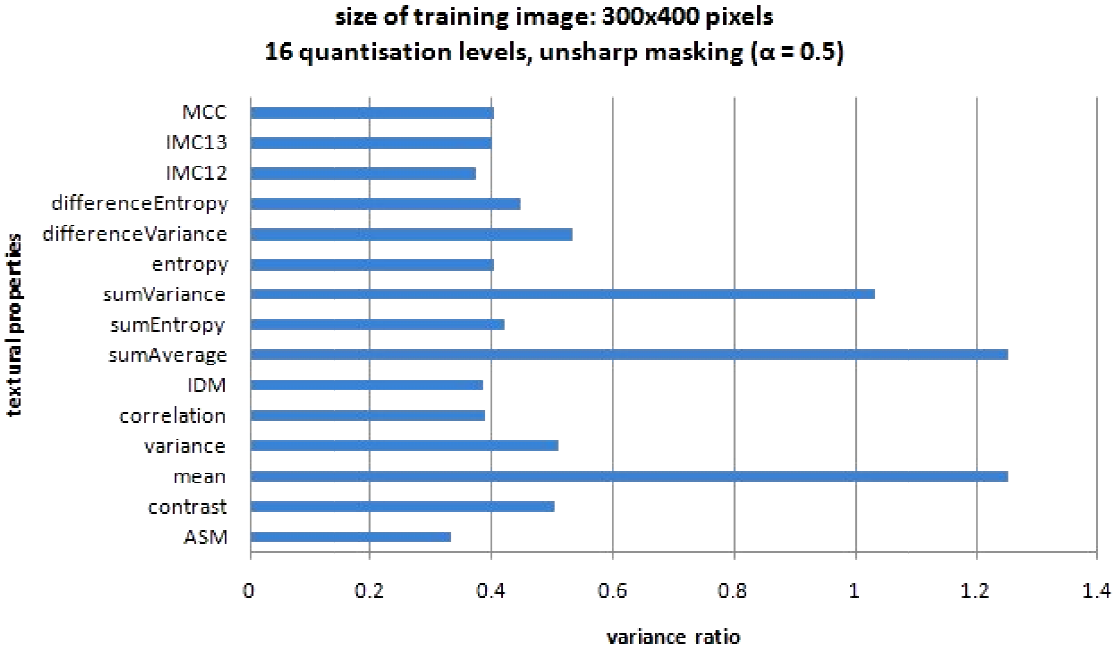


Figure 4.10 Variance ratio bar graph for training images with 300x400 pixels image size, 16 quantisation levels, and unsharp masking coefficient α = 0.5

It is clear from Table 4.1 and Figures 4.2 to 4.10 that only the histogram equalisation process had a significant effect on the variance ratios of the textural properties of the training images. Based on the bar graphs in Figure 4.2 to Figure 4.10, except for Figure 4.9 , three properties dominate in terms of variance ratio: the mean, sum average, and sum variance. Following Boland *et al.* (1998), these three properties exhibit "good" characteristics that widely separate the classes from each other while simultaneously keeping the individual classes tightly clustered. The bar graphs in Figure 4.2 to Figure 4.10, except for Figure 4.9, also support the idea of resizing the original images to 300 x 400 pixels and show that the quantisation level is not particularly important. Since data was already available using 16 quantisation levels in the previous execution of algorithms, data using 8 quantisation levels was not considered for the classification stage since, as mentioned, quantisation level does not affect the variance ratios. Figure 4.9 illustrates that the process of histogram equalisation destroys image information vital to texture analysis. This should come as

no surprise since histogram equalisation practically reorders the pixel information in an image. The last bar graph, Figure 4.10, shows that edge enhancement is not necessary even though it certainly can enhance the appearance of an image for human viewing. Unlike histogram equalisation though, unsharp masking preserves the textural information in an image.

4.3 Feature Selection Using Genetic Algorithm and Kohonen Self-Organising Map

This part of the study explores the idea of using genetic algorithm or GA to select the most discriminating features among the 15 considered in the previous section, section 4.2, without using the known classes of the training images. Stated in another way, the idea is to use GA to implement feature selection in an unsupervised manner by using the map error of a Kohonen self-organising map (KSOM) as the fitness function. There are two commonly used KSOM error terms (Uriarte and Martin, 2005): quantisation error and topographic error. Only the quantisation error has been used here to quantify the quality of a Kohonen map. The GA algorithm optimises the Kohonen map by selecting from populations of coefficients to each of the 15 prospective textural properties a set of values that will yield the minimum quantisation error. Although the coefficients can take any real number, the possible values are zero and non-zero only. The reason for this is that all inputs into the Kohonen map in this study are normalised by transforming the values into standard values with zero mean and unity variance. This process makes the effective values for the coefficients practically binary by taking only the absolute values of all non-zero coefficients. A zero coefficient means that the corresponding property that resulted in it should be eliminated to get an optimum Kohonen map while a non-zero coefficient means that the associated property is important for an optimum Kohonen map. Figure 4.11 is a

schematic diagram showing the interaction between the GA and the KSOM. The Kohonen map was fixed to have 200 neurons.

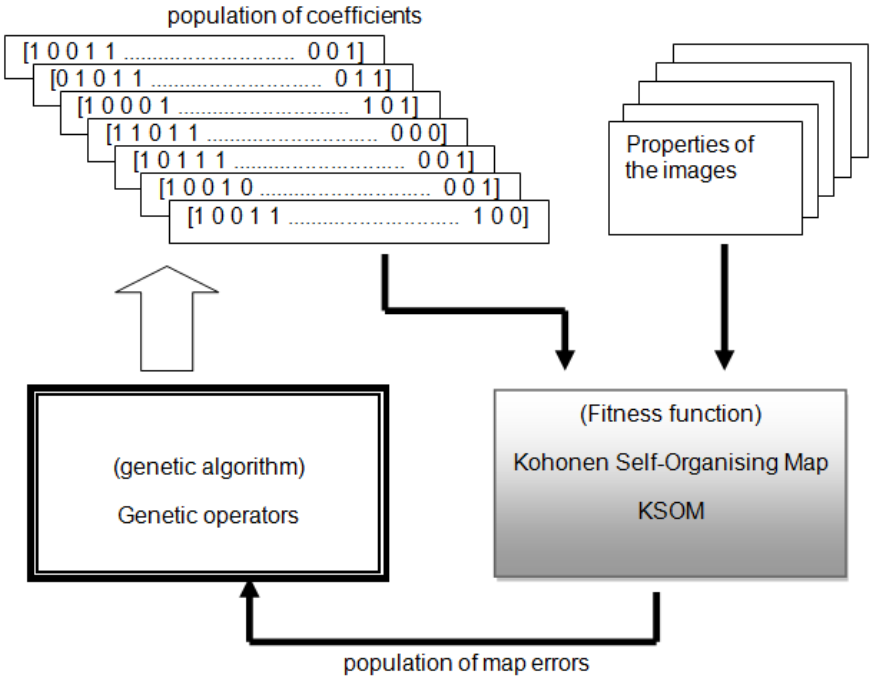


Figure 4.11 Schematic diagram of the GA-KSOM feature selector

One of the major obstacles in using the quantisation error in the fitness function is the fact that its value becomes zero when all the coefficients are selected to be zero. In this case, a trivial solution is produced and terribly affects the course of the genetic evolution of the population of coefficients. Another factor that can destroy the genetic evolution process is the possibility of having no coefficient equal to zero which simply means that all coefficients are important and must be selected. This is obviously another trivial output since one of the aims of feature selection is to reduce the number of inputs to a classifier system and therefore not all of the 15 prospective features must be used. It is also very unlikely that all 15 features are all equally good discriminators. To solve this problem, a penalty function in the fitness function has been devised in this study. The idea is to introduce a function that increases the error by adding some value to it whenever the variance of the set of coefficients is small. Addition is better than

multiplication in this case since a zero value in either the map error term or the penalty function term is possible. If a zero value occurs in either of these two terms, then if multiplication was used instead of addition, the fitness function would give a false zero value. In this way with the use of the addition operation, the fitness function is modified to give 'good' values only when there are a few properties with non-zero coefficients while at the same time avoid an all-zero set of coefficients. Figure 4.12 shows a plot of the penalty function operating within the GA fitness function.

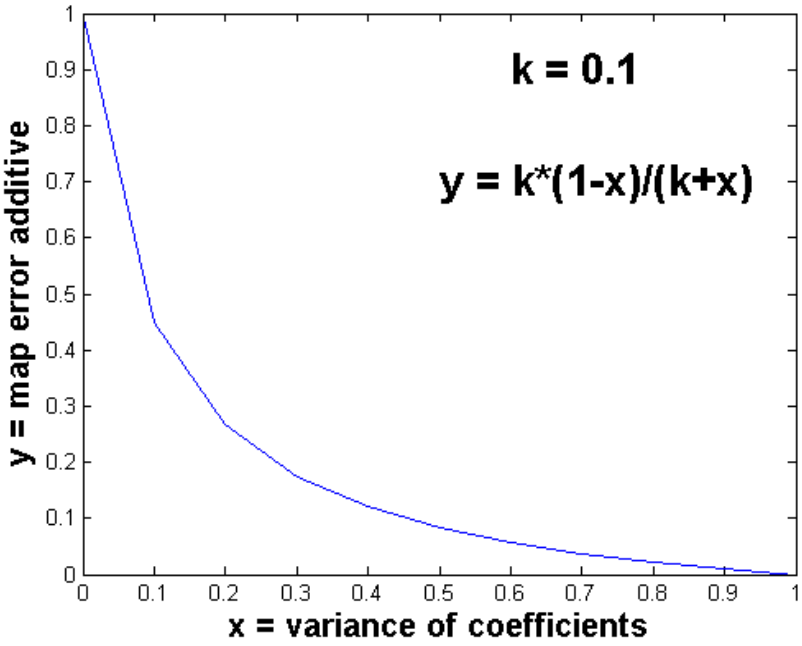


Figure 4.12 Penalty function within the GA fitness function

The function depicted in Figure 4.12 is actually a reversed half sigmoid function. The parameter k determines the curvature of the plot. The value of k used in this study was 0.1 which was heuristically obtained. Other values very close to 0.1 also gave somewhat similar results. As a parameter, large values of k cause the plot to become a straight line, while small values close to zero make the curve increasingly sharp.

Table 4.2 summarises the parameters and settings used in the implementation of the genetic algorithm to search for the optimum set of feature coefficients. Except for the number of variables, the rest of the parameters were heuristically set. The

maximum number of generations was limited to 25 since it was observed that at this value, settling of the fitness function output was already apparent. The following is a list of steps that were undertaken to implement the GA search of coefficients using KSOM with penalty function as fitness function:

- Generate the textural properties of all the training images and store results in a *.data file. Use the MATLAB script program "image2FeatureDATAFile.m" for this task.
- Run the MATLAB GA Toolbox using "glf_SOMFitnessFunction.m" as the fitness function
- Produce horizontal bar graphs to visualise the results

Table 4.2 Parameters and settings used in the implementation of GA

Number of Variables	15
Population size	30
Population type	Double vector
Maximum number of generations	25
Selection function	Roulette wheel
Elite Count	2
Crossover Fraction	0.8
Initial Population	random
Fitness Scaling Function	Fit scaling rank
Crossover Function	scattered

Table 4.3 Table of values of feature coefficients obtained from GA-KSOM search algorithm with elapsed times and best (minimum) fitness values

	Run #1	**Run #2**	**Run #3**	**Run #4**
[1]ASM	0	0	0	0
contrast	2.181	1.7667	0	2.5059
mean	0	0	0	0
variance	0	0	0	0
correlation	0	0	0	0
[2]IDM	0	1.7397	0	1.6358
sumAverage	0	0	1.0091	0
sumEntropy	0	0	0	0
sumVariance	0	1.4352	0	0
entropy	0.405	0	0	0
differenceVariance	1.456	0	0	1.0155
differenceEntropy	0	0	2.9536	0
[3]IMC12	0	0	0	0

[3]IMC13	0	0	0	0
[4]MCC	0	0	0	0
[1] Angular second moment [2] Inverse Difference Moment [3] Information Measures of Correlation (f12 and f13) [4] Maximal Correlation Coefficient				
Elapsed time [min:sec]	09:15	05:46	05:11	07:49
Final best fitness	0.0401	0.0599	0.0201	0.0136

The results of the GA-KSOM search are shown in Table 4.3 and Figures 4.13 to 4.20. These results consist of a table of values and bar graphs of feature coefficients from four (4) runs of the GA-KSOM search algorithm.

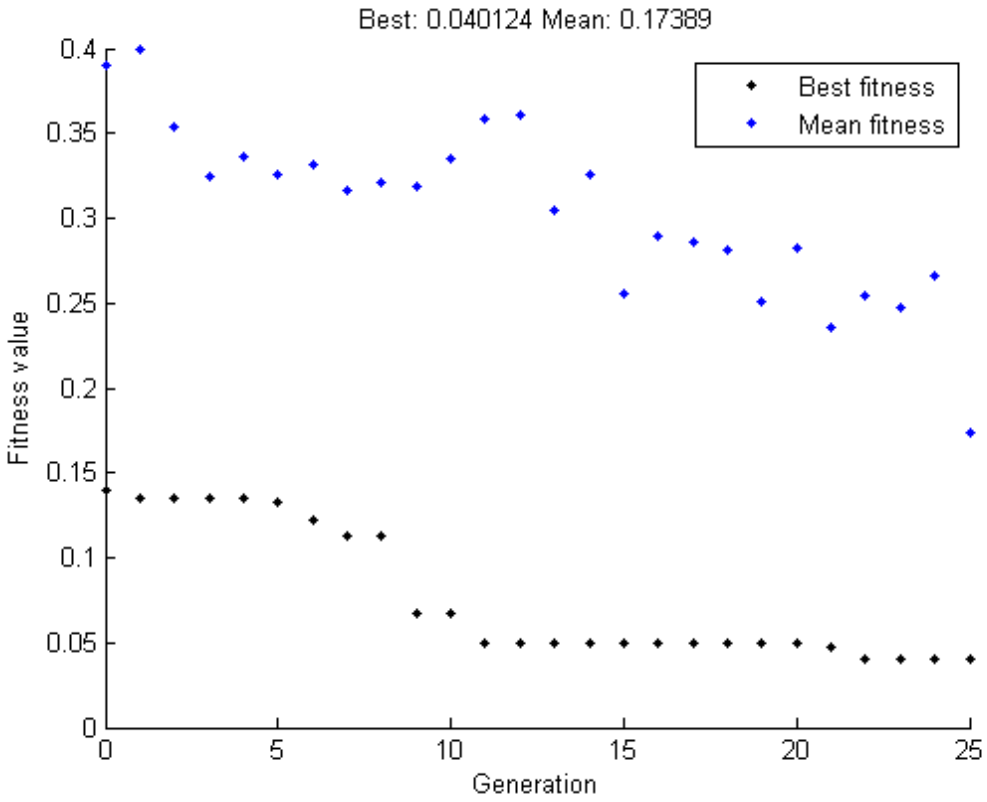


Figure 4.13 Fitness values for run #1 of the GA-KSOM search algorithm

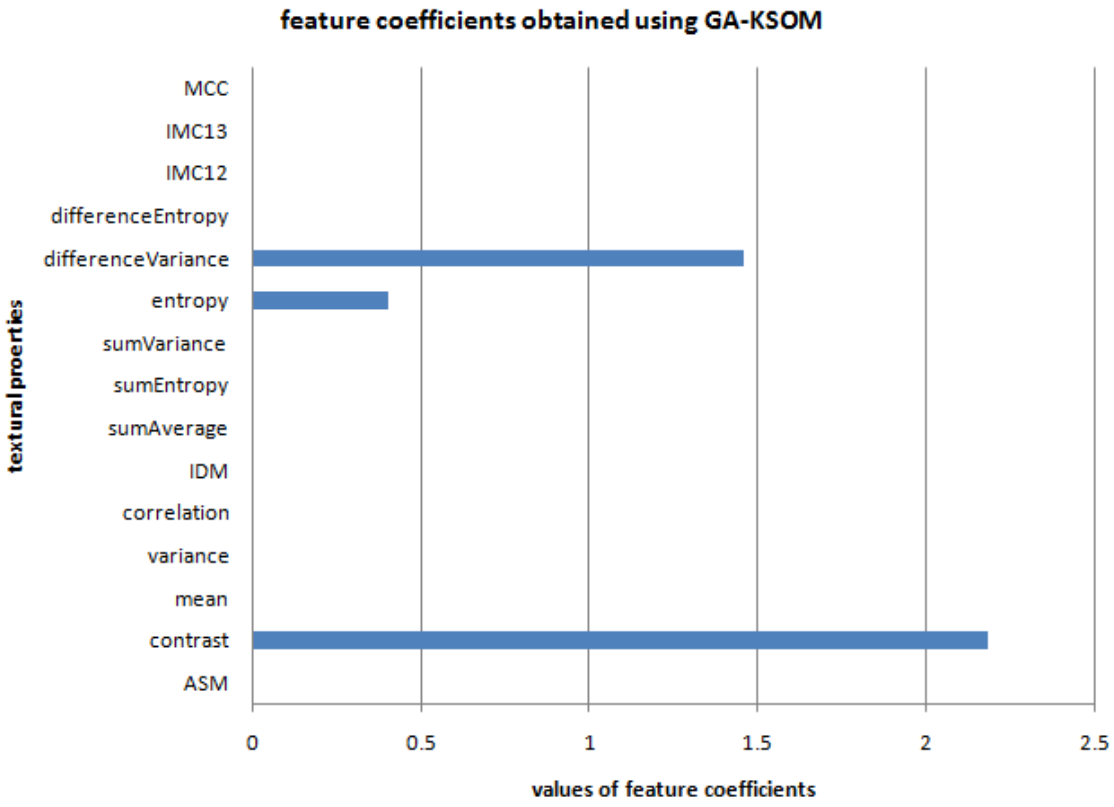


Figure 4.14 Feature coefficients from run #1 of the GA-KSOM search algorithm

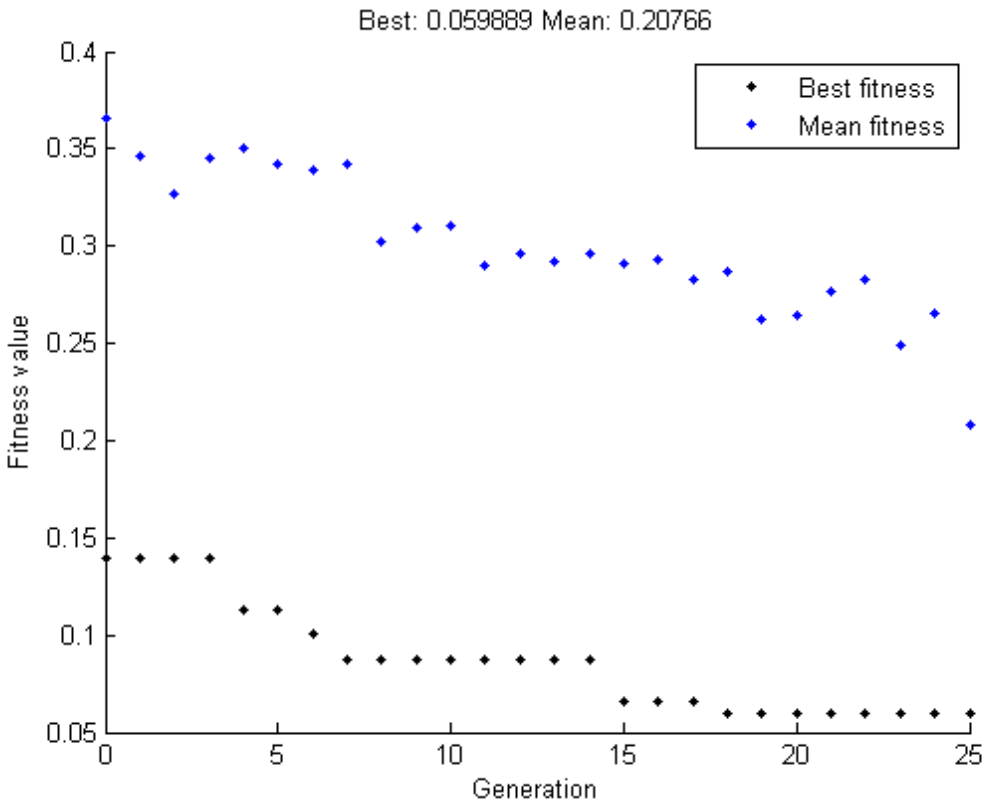


Figure 4.15 Fitness values for run #2 of the GA-KSOM search algorithm

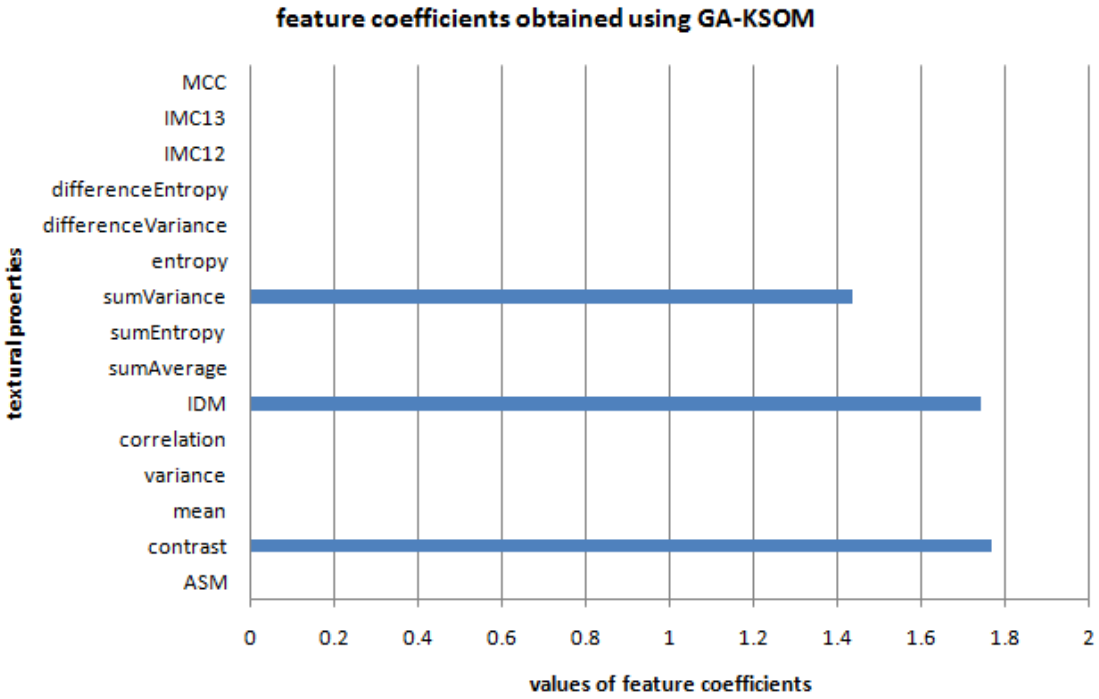


Figure 4.16 Feature coefficients from run #2 of the GA-KSOM search algorithm

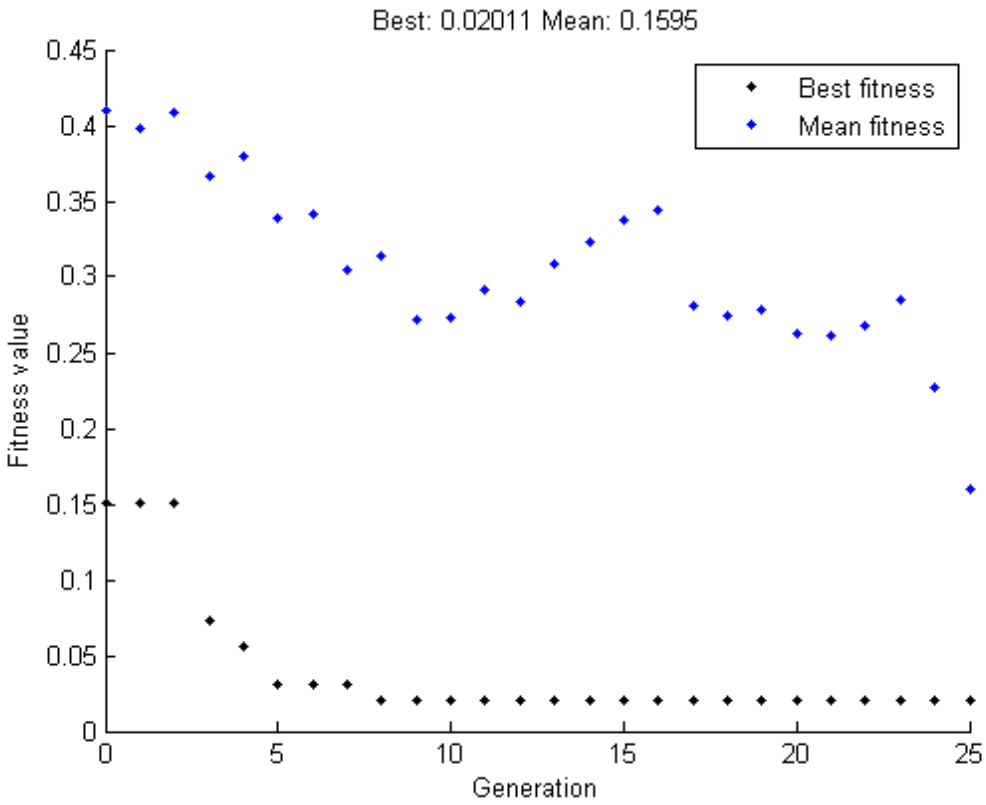


Figure 4.17 Fitness values for run #3 of the GA-KSOM search algorithm

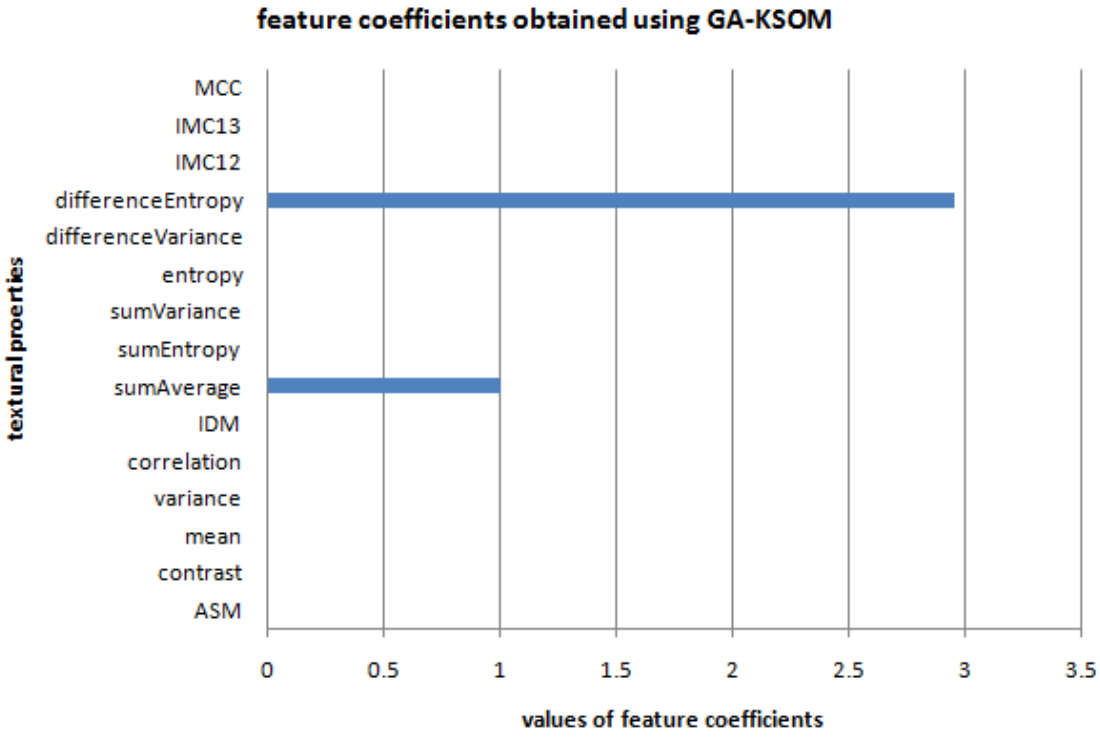


Figure 4.18 Feature coefficients from run #3 of the GA-KSOM search algorithm

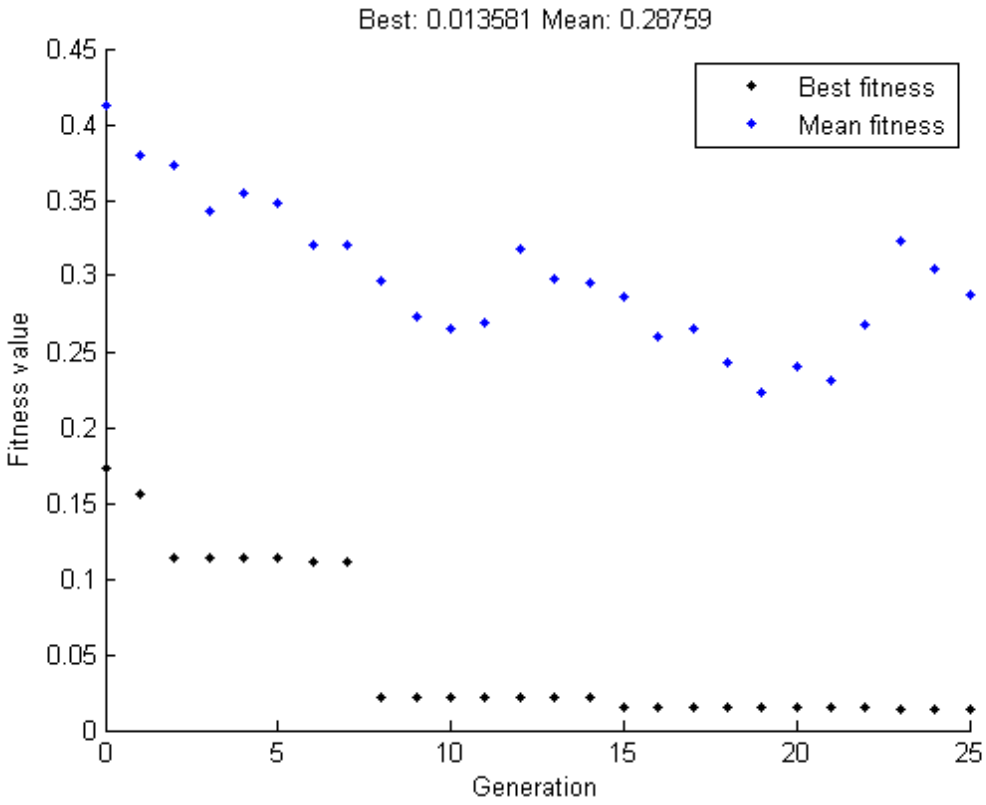


Figure 4.19 Fitness values for run #4 of the GA-KSOM search algorithm

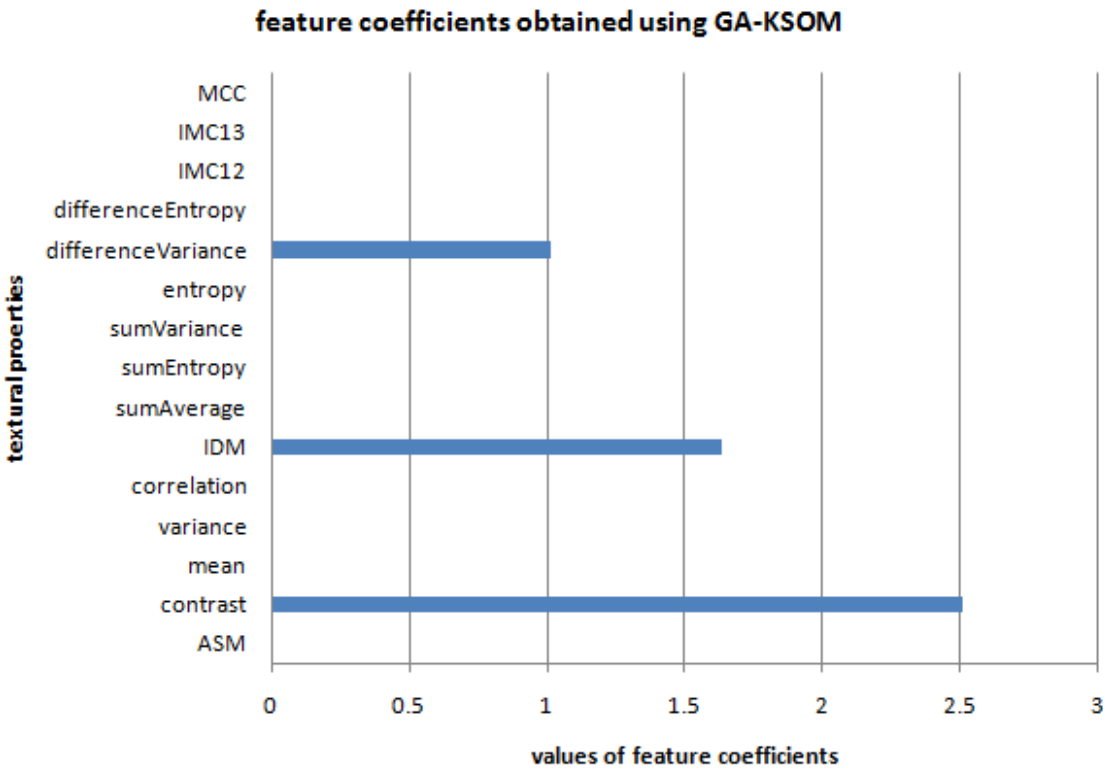


Figure 4.20 Feature coefficients from run #4 of the GA-KSOM search algorithm

A peculiar outcome of the results is that the graphs are different from each other although they have some similarities wherein some features have been picked more than once. This indicates that the global optimum may or may not have been found. Actually, the 4 results are representative of the numerous runs of the GA-KSOM algorithm where similar consecutive results were not obtained. This could mean any of the following:

- The map error used in the fitness function needs to be modified or changed with another parameter,
- The penalty function idea is not an appropriate fix to the problem of getting trivial outputs from the map error function.

In any case, the final judge as to which set of properties was selected to be the 'best' is up to the classifier in the next chapter.

4.4 Preparing for Image Classification

The use of the variance ratio suggests that there are three (3) textural properties that differentiate themselves from others as far as class discrimination power or effectiveness is concerned. These are mean, sum average, and sum variance. Both the mean and sum average appear to be equal in terms of their discriminating power while the sum variance is also not far behind. The results of the feature selection/analysis using genetic algorithm and Kohonen Self-Organising Map or GA-KSOM tell a different story. Unlike in the use of the variance ratios where only the histogram equalization appeared to have made a significant difference in the bar graphs, consistent sets of coefficients were not obtained during the numerous executions of the GA-KSOM search algorithm. This might suggest that the feature selection problem in this case might be multi-modal. In other words, global optimum was not achieved either because it was impossible to obtain or a better fitness function is needed. The 'arbitrariness' in the results of the GA-KSOM algorithm generally only referred to specific combinations of features. Some features however were appearing more frequently than others during several executions of the algorithm. This observation suggests that some of the features are more discriminating than others. Whatever the case may be, the classification of images in Chapter 5 will provide judgement as to which feature selection process is more effective.

To summarise the findings in this chapter, the following is a list of features that will be used and evaluated by the neuro-fuzzy classifier in Chapter 5:

Set A - Mean, Sum Average, Sum Variance (from variance ratio analysis)

Set B - Mean and Sum Average (from variance ratio analysis)

Set C - Contrast, Entropy, and Difference Variance (from GA-KSOM)

Set D - Contrast, Inverse Difference Moment or IDM, and Sum Variance (from GA-KSOM)

Set E - Sum Average and Difference Entropy (from GA-KSOM)

Set F - Contrast, Inverse Difference Moment or IDM, and Difference Variance (from GA-KSOM)

Chapter 5 – AUTOMATIC CLASSIFICATION OF IMAGES

The methods used in the previous chapter on feature selection yielded a number of interesting sets of features. The true test for any set of features put forward by any feature selection process is the classification itself of training and test images. This will ultimately tell us which features and/or feature combinations work. In this chapter, therefore, the objective is to evaluate the sets of features that have been suggested at the end of Chapter 4 and the list is repeated here for convenience:

Set A - Mean, Sum Average, Sum variance (from variance ratio analysis)

Set B - Mean and Sum Average (from variance ratio analysis)

Set C - Contrast, Entropy, and Difference Variance (from GA-KSOM)

Set D - Contrast, Inverse Difference Moment or IDM, and Sum Variance (from GA-KSOM)

Set E - Sum Average and Difference Entropy (from GA-KSOM)

Set F - Contrast, Inverse Difference Moment or IDM, and Difference Variance (from GA-KSOM)

5.1 Mean Relative Difference Confusion Matrix (MRDCM) and Classification Performance Index (CPI)

One of the problems identified in this study is the need to devise classifier performance metrics that highlight not only the classification accuracy but also the areas for improvement of a classifier under consideration. A commonly used tool to examine the performance of a classifier is the confusion matrix which is a table of numbers of correct classifications and misclassifications. If one wants to simply produce a single number out of the confusion matrix as a measure of classification

performance, the sum of the diagonals of the matrix is usually chosen and normalised to produce what is called the percent accuracy. The problem with this performance parameter is that it does not show the gravity of mistakes committed by the classifier in problems with more than two classes. For example, in this study where there are three classes of images: normal, adenomatous polyp, and cancerous cases, the percent accuracy will not yield information as to whether a cancerous case was misclassified as normal or as adenomatous polyp. Clearly in "human" logic, it is less of a mistake to classify a cancerous case as adenomatous polyp than to classify it as normal. Erroneous downgrading from cancerous to normal can lead to a serious case not given enough scrutiny and is therefore the worst mistake that can be made by a classifier. As for the confusion matrix, although it is in itself an excellent tool to analyse the performance of a classifier, it is not directly compatible with the output of ANFIS. The confusion matrix tabulates the counts (whole numbers) of classifications and misclassifications while ANFIS, since it is a Sugeno-type fuzzy inference system or FIS, generally gives out real numbers. There are two alternatives that can be adopted to fix this. One is to introduce threshold values for the output of ANFIS and the other is to devise another classification performance matrix which can "handle" the ANFIS output values. The latter choice is more preferred in this study because it has the advantage of maintaining the spectral nature of histopathologic image classification and characterisation. It is believed in this study that this approach is closer to how human pathologists view this kind of problem. Therefore, in this research, a new classification performance matrix, called the MRDCM, is proposed. The MRDCM, which stands for *Mean Relative Difference Confusion Matrix*, tabulates the average differences of classification output values of the images and three constants defined by the following:

0.0 – for normal case

0.5 – for adenomatous polyp case, and

1.0 – for cancerous case.

Table 5.1 shows the general format of an MRDCM or *Mean Relative Difference Confusion Matrix.* Unlike the usual confusion matrix, the main diagonal elements of an MRDCM are ideally zero or close to zero since it is desired that the classification of the images should be correct and therefore have very small, if not zero, average differences with the ideal ANFIS output value for each case. For the off-diagonal elements, it is desirable to have non-zero values close to 0.5 or 1.0.

Table 5.1 General format of an MRDCM or *Mean Relative Difference Confusion Matrix*. The elements a, e, and i are the main diagonal elements. The rest of the elements are the off-diagonal elements.

	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	a	b	c
Predicted Aden. Polyp	d	e	f
Predicted Cancerous	g	h	i

Each element in the matrix can be expressed as:

$$\boldsymbol{x_{ij}} = \frac{\sum_{k=1}^{n_j} |o_j(k) - c_i(k)|}{n_j} \qquad \textbf{Equation 5.1}$$

where

x_{ij} = element in the matrix at row *i* and column *j*

$o_j(k)$ = ANFIS output value for image *k* at class *j*

n_j = total number of images in class *j*

$$c_i(k) = \begin{cases} 0.0 & if\ i = 1 \\ 0.5 & if\ i = 2 \\ 1.0 & if\ i = 3 \end{cases}$$

In optimising classifiers, it would be very advantageous to be able to express the performance of a classifier into a single number or scalar just like the percent accuracy of a confusion matrix. As pointed out earlier, the percent accuracy parameter does not

take into account the gravity of the misclassifications of a classifier for problems with more than 2 cases. The new idea that is being proposed in this study is to introduce a parameter called the Classification Performance Index or CPI that precisely brings with it the information conveyed by percent accuracy plus additional measures of classification failures. The CPI metric is arrived at by first calculating the corresponding confusion matrix using threshold values for the adenomatous polyp and cancerous cases and normalising the elements by using the sum of elements per column or class as divisor. Next, the confusion matrix with normalised elements is then multiplied element-wise by a new matrix referred to here as factor matrix, which is essentially a weight matrix. The product, which is sometimes referred to as Hadamard or Schur product in matrix multiplication, is another matrix similar in size to the confusion matrix and the factor matrix. The factor matrix contains elements that act as multipliers similar to connection weights in a feed-forward neural network. Finally, the CPI parameter is calculated as the sum of all the elements of the element-wise product of the normalised confusion matrix and the factor matrix. The idea behind the factor matrix is to select specific real numbers as elements that will seek proportional contributions of the specific elements of the confusion matrix to the CPI parameter. In order to make the CPI reflect the failure-to-success spectrum of a classifier, the entries in the factor matrix must be selected to get more contribution from the successes and less from the failures in the numbers tabulated in the confusion matrix. This was accomplished in this study by suggesting a ranking of the elements of the confusion matrix according to the degree of success and gravity of failure of the classifier expressed as a set of multiplying factors. Tables 5.2, 5.3, and 5.4 show the format of the confusion matrix used in this study, the format of the factor matrix, and the suggested ranking of the corresponding elements according to a set of multiplying factors, respectively.

Table 5.2 Format of the confusion matrix used in this study.

	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	A	B	C
Predicted Aden. Polyp	D	E	F
Predicted Cancerous	G	H	I

Table 5.3 Format of the factor matrix. The letters assigned to each element of the matrix correspond to the left column of Table 5.4 and to the entries in Table 5.2 as multipliers.

a	b	c
d	e	f
g	h	i

Table 5.4 The suggested ranking of the elements of the factor matrix with the multiplying factors. Match the letters on the left column to the entries in Table 5.3.

Location in the factor matrix	Multiplying factor
i	+1/3
e	+1/3
a	+1/3
d	-0.05
h	-0.1
g	-0.2
b	-0.3
f	-0.4
c	-0.5

It can be observed that the multiplying factors in Table 5.4 together produce an effect on the CPI wherein the positive and negative factors counteract each other when

multiplied by the confusion matrix. The entries i, e, and a get +1/3 each since the numbers in these locations in the confusion matrix represent the correct classifications. Their factors have been purposely chosen to sum-up to 1.0 or 100% because they represent the perfect score. The rest of the entries are all assigned negative factors representing a penalty against the CPI since they are the multipliers of the off-diagonal elements of the confusion matrix. It can be observed that the factors in entries c, g, and b all sum-up to -1.0 or -100% which is considered to be the exact opposite of a perfect score in classification in this study. Entry c is assigned the greatest penalty effect since it corresponds to the worst possible mistake that can be committed by a classifier which is a misclassification of cancer into normal. Since entry f is considered as between entries c and b, therefore c = -0.5, b = -0.3, g = -0.2 and f = -0.4. Entry d is considered here as the element in the factor matrix that corresponds to the least serious misclassification wherein a truly normal case is classified as adenomatous polyp by mistake while entry h had to be just worse than entry d. With g = -0.2 and a = 1/3, therefore entries h and d had to assume -0.1 and -0.05 values, respectively. Therefore, Table 5.4 suggests that the factor matrix should be expressed as in equation 5.2.

$$FM = \begin{bmatrix} +1/3 & -0.3 & -0.5 \\ -0.05 & +1/3 & -0.4 \\ -0.2 & -0.1 & +1/3 \end{bmatrix}$$ **where: FM = factor matrix** **Equation 5.2**

Putting it all together now, the CPI can be calculated by first getting the entry-wise product of the confusion matrix and the factor matrix, and then obtaining the sum of all the elements of the resulting matrix. Mathematically, for 3 classes this can be expressed as:

$$CPI = \sum_{i=1}^{3}\sum_{j=1}^{3}\frac{CM_{ij}FM_{ij}}{N_j}$$ **Equation 5.3**

where :

CPI = classification performance index

CM_{ij} = entry in the confusion matrix at row i and column j

FM_{ij} = entry in the factor matrix at row i and column j

N_j = total number of elements in class or column j

5.2 Implementation of ANFIS

The chosen classifier in this study is the ANFIS which stands for *Adaptive-Network-based Fuzzy Inference System* or semantically equivalently *Adaptive Neuro-Fuzzy Inference System*. It is a hybrid neuro-fuzzy system proposed by J-S Jang (1993) and uses only the Sugeno-type of fuzzy system. This classifier is well-suited for this study for the following reasons:

1. the output can be made to be a single real number which can range from 0 to 1 with 0 representing a normal case, a value of 1 representing a cancerous case, while the real numbers in between represent the varying degrees of dysplasia,
2. the input is the training data where "knowledge" is to be extracted, and
3. ANFIS uses a hybrid learning procedure which converges much faster than using just the back-propagation training scheme (Jang, 1993).

The following steps summarise the procedure employed in this study to classify the training and testing images using the sets of features suggested in Chapter 4:

1. If the *.data files for the training and test images do not exist yet, use the MATLAB script program "image2FeatureDATAFile.m" to produce them.
2. Generate the training and test *.dat files from the *.data files using the MATLAB program "writeToFileChosenPropertiesForANFIS.m". [The main difference between a *.dat file and a *.data file is that a *.dat file only contains the properties that were selected in the feature selection process.]

3. Run the program “ANFISthesisImplementationCommandLine.m” to implement ANFIS and produce the necessary classification results.

All the codes mentioned in the procedure just enumerated are included in the Appendix sections A.3, A.6 and A.7. The confusion matrices were obtained using 0.25 and 0.75 as threshold values for adenomatous polyp and cancerous cases, respectively. These values were chosen since the output and input mapping are assumed to be linear with the use of Sugeno FIS in the ANFIS classifier. Since the main values were 1.0, 0.5 and 0.0, it is natural to use middle values for the thresholds. ANFIS outputs that fell below 0.25 were considered to be classified as normal while ANFIS outputs that fell between 0.25 and 0.75 were classified as adenomatous polyp. ANFIS outputs above 0.75 were considered to be cancerous. The following are the results of the implementation of the ANFIS classifier on the feature sets (sets A to F) given in Chapter 4.

Set A feature combination [Mean, Sum average, and Sum variance]:

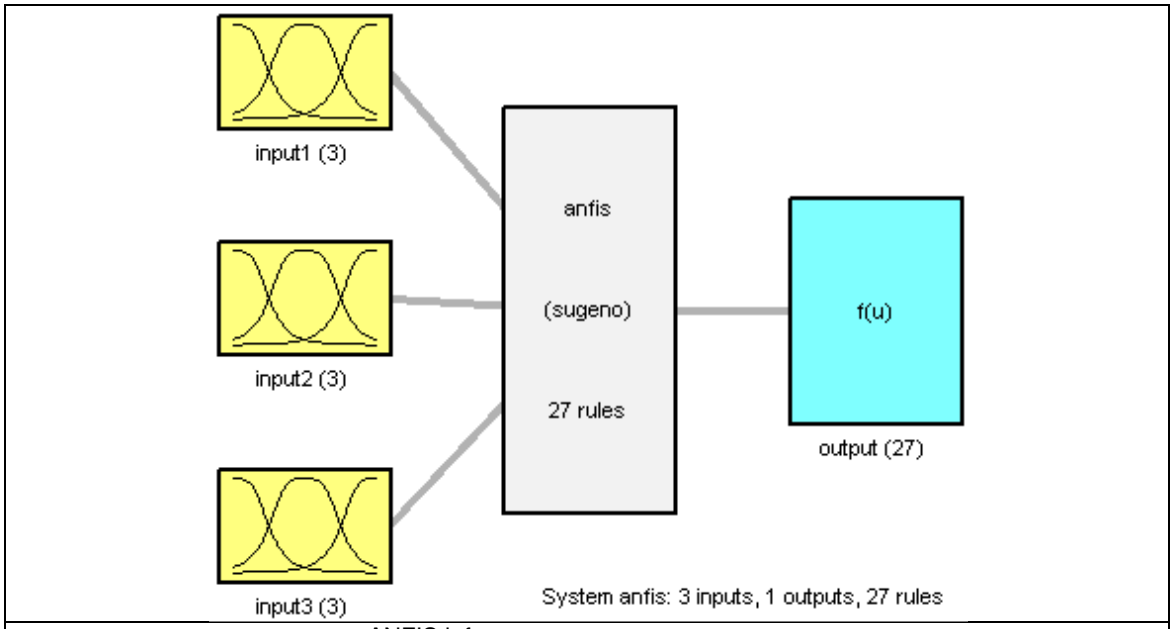


Figure 5.1 ANFIS Structure of Set A features

Figure 5.1 shows the topological arrangement of the input and output variables of the ANFIS using Set A features. It can be noticed from Figure 5.1 that 3 membership functions were used for each input. Figure 5.2 shows that the membership functions were not affected during the training process.

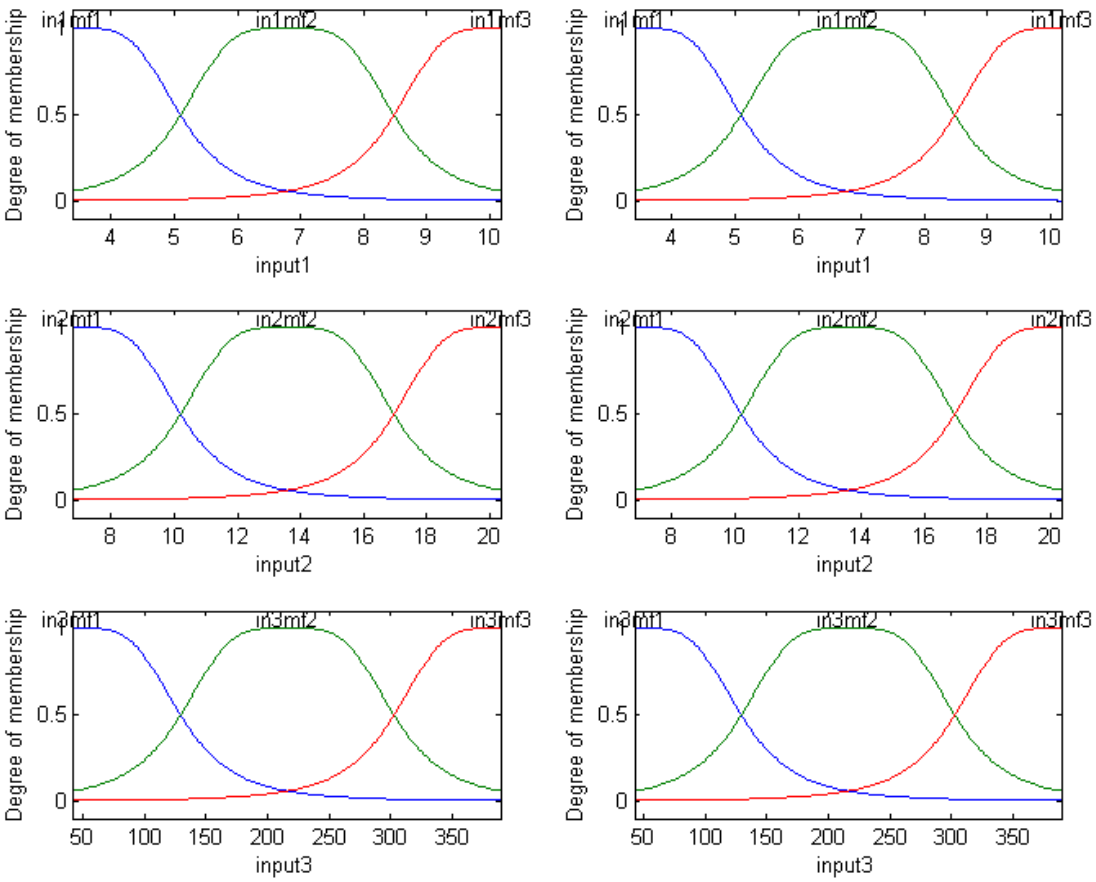


Figure 5.2 ANFIS Membership Functions using Set A features: Mean (input1), Sum average (input2), Sum variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

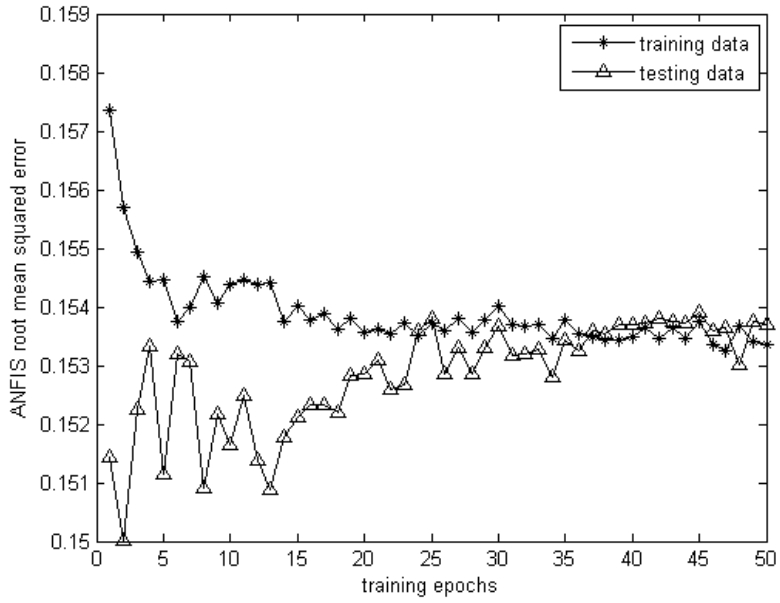


Figure 5.3 ANFIS root mean squared errors during training for Set A features: Mean, Sum average, Sum variance

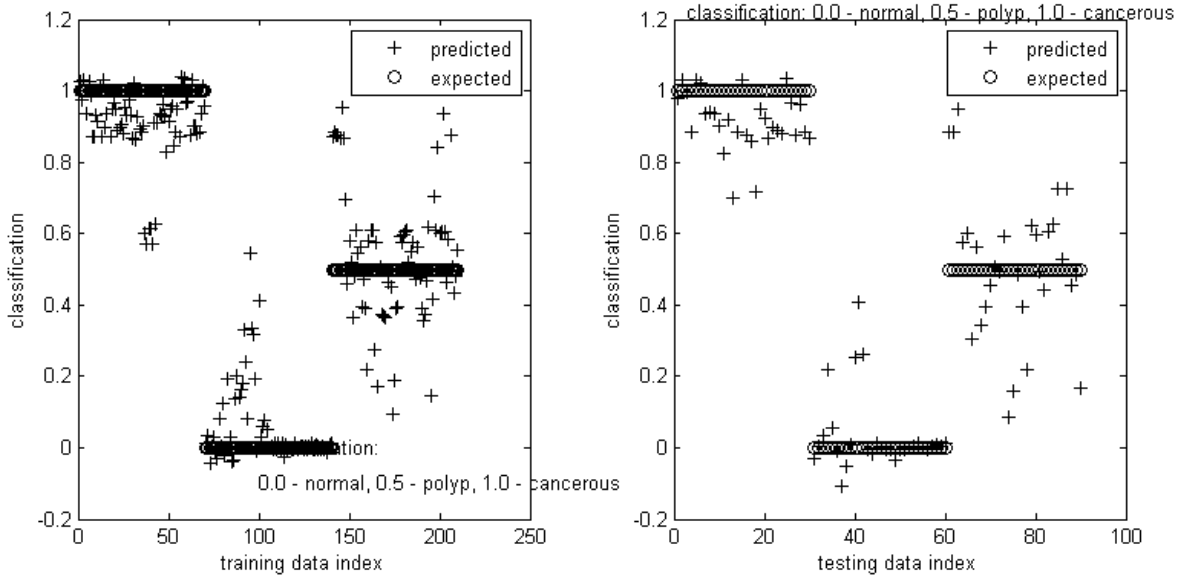


Figure 5.4 Classification performance trained ANFIS using training and testing data sets for Set A features: Mean, Sum average, Sum variance

The root mean squared errors during the training of the ANFIS using Set A are shown on Figure 5.3. On Figure 5.4, the clustering of ANFIS outputs using Set A features is presented for both training and testing images. It can be observed from both Figure 5.4 and Figure 5.5 that the polyp case is the most difficult to classify. The

effectiveness of the ANFIS classifier using Set A features is tabulated in Table 5.5 and Table 5.6.

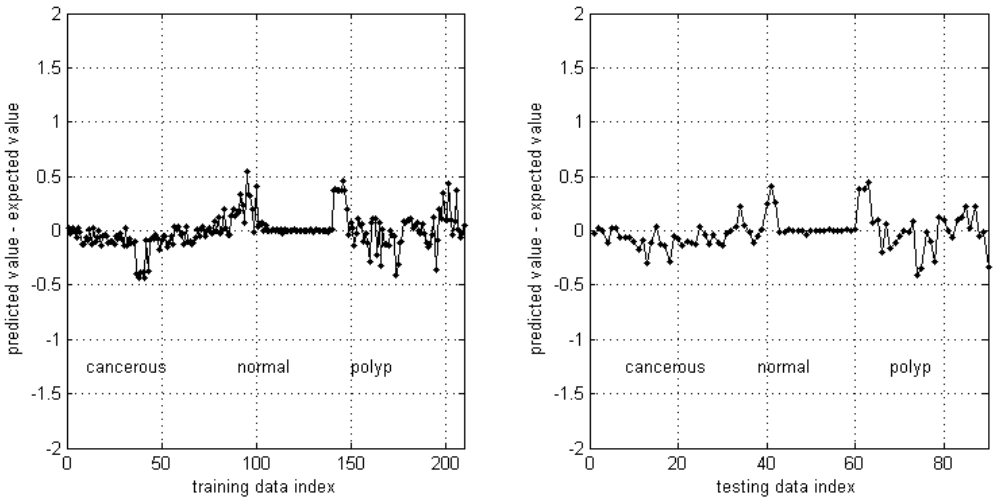


Figure 5.5 Classification Difference of Trained ANFIS using training and testing data sets for Set A features: Mean, Sum average, Sum variance

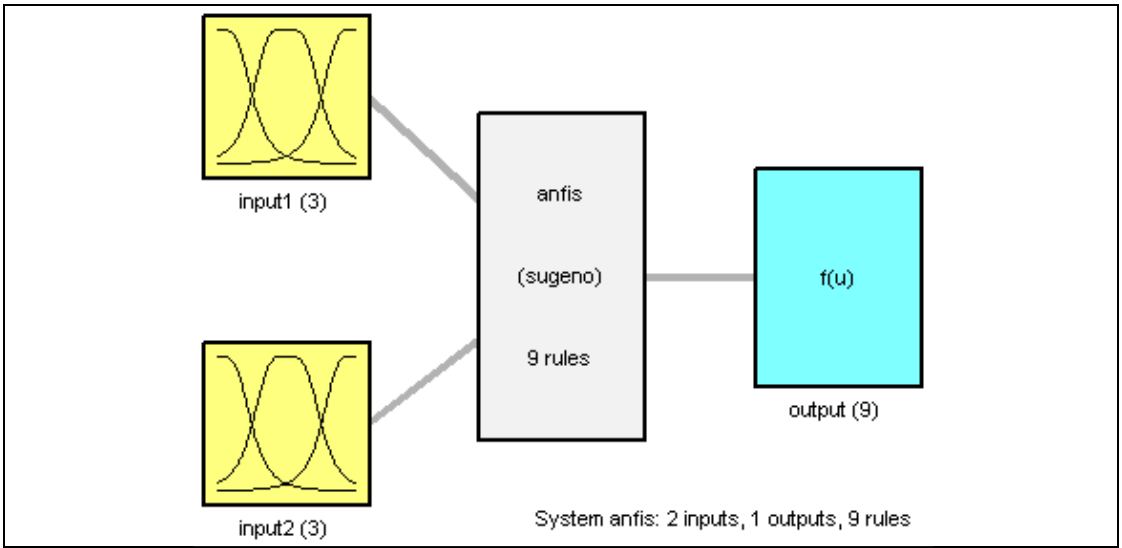
Table 5.5 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set A features: Mean, Sum average, Sum variance

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.0843	0.7165	1.2107	0.0704	0.6826	1.2206
Predicted Aden. Polyp	0.5957	0.1906	0.544	0.6213	0.2046	0.554
Predicted Cancerous	1.2605	0.6168	0.1336	1.288	0.6507	0.1257

Table 5.6 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set A features: Mean, Sum average, Sum variance with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	65	5	0	27	4	0
Predicted Aden. Polyp	5	55	6	3	23	2
Predicted Cancerous	0	10	64	0	3	28
Percent Accuracy	87.6190%			86.6667%		
Classification Performance Index	0.8026			0.7850		

Set B feature combination [Mean and Sum average]:



ANFIS info:
Number of nodes: 35
Number of linear parameters: 27
Number of nonlinear parameters: 18
Total number of parameters: 45
Number of training data pairs: 210
Number of checking data pairs: 90
Number of fuzzy rules: 9

Input(1) = mean
Input(2) = Sum Average

Figure 5.6 ANFIS Structure of Set B features

Figure 5.6 shows the topological arrangement of the input and output variables of the ANFIS using Set B features. It can be noticed from Figure 5.6 that 2 membership functions were used for each input. Figure 5.7 shows that the membership functions were not affected during the training process.

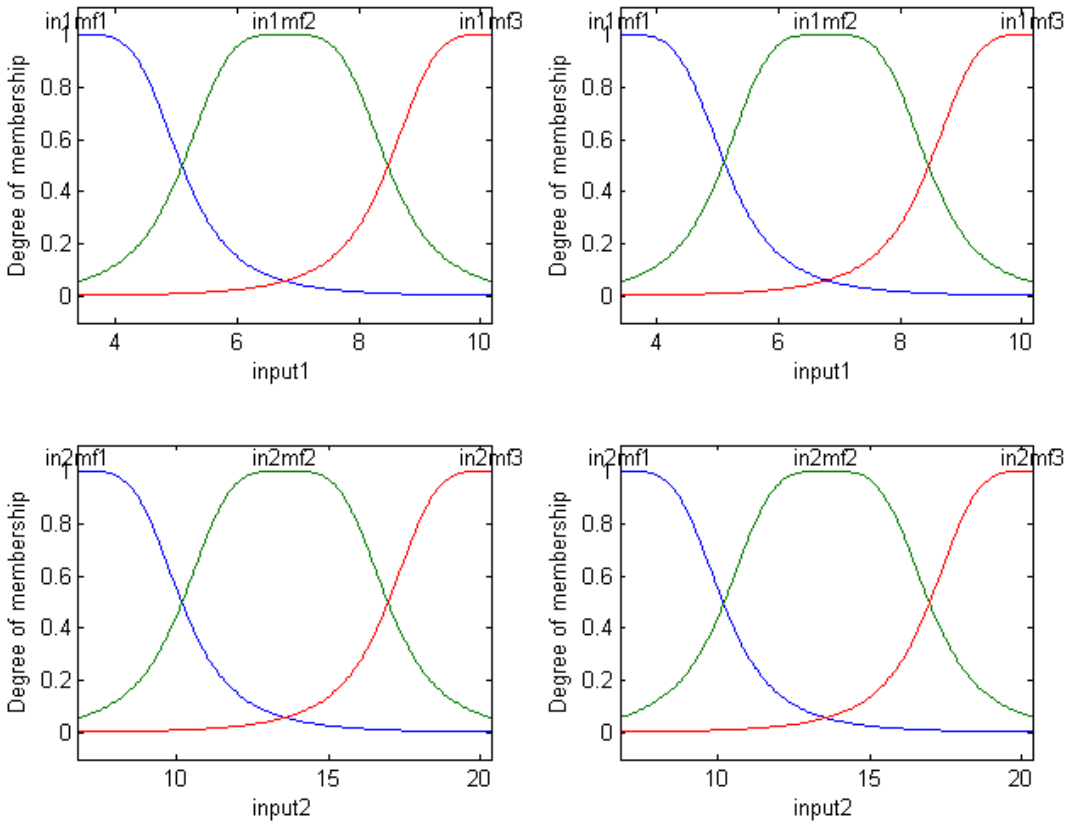


Figure 5.7 ANFIS Membership Functions using Set B features: Mean (input1), Sum average (input2). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

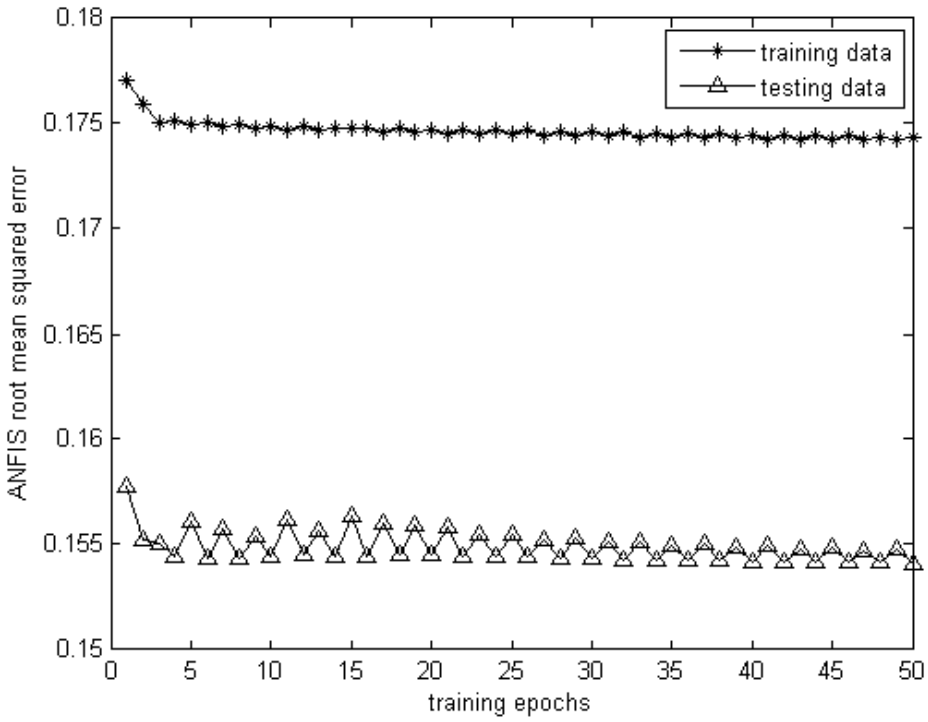


Figure 5.8 ANFIS root mean squared errors during training for Set B features: Mean and Sum average

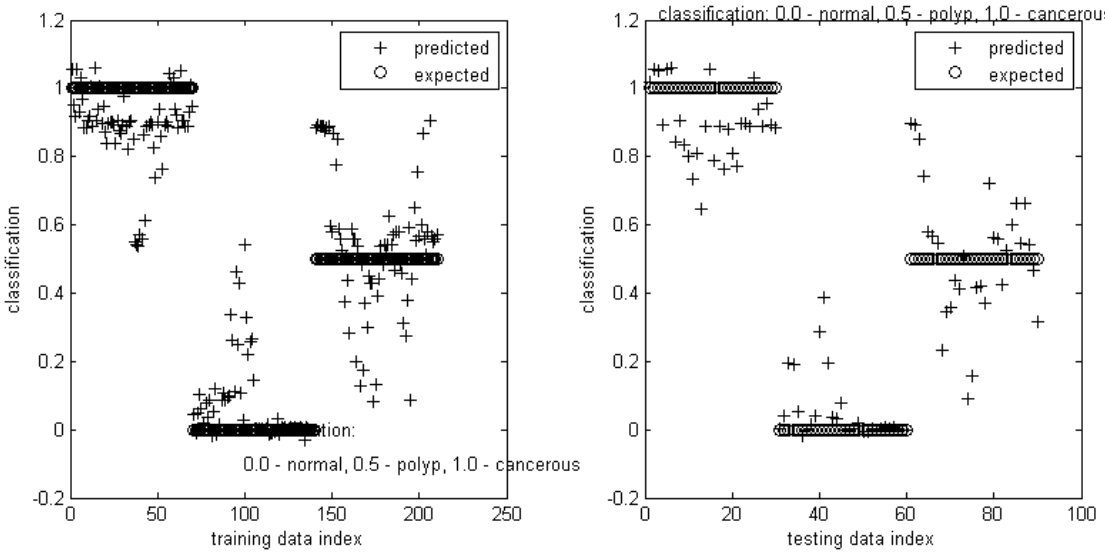


Figure 5.9 Classification performance trained ANFIS using training and testing data sets for Set B features: Mean and Sum average

The root mean squared errors during the training of the ANFIS are shown on Figure 5.8. On Figure 5.9, the clustering of ANFIS outputs using Set B features is presented for both training and testing images. It can be observed from both Figure 5.9 and Figure 5.10 that the polyp case is the most difficult to classify. The effectiveness of the ANFIS classifier using Set B features is tabulated in Table 5.7 and Table 5.8.

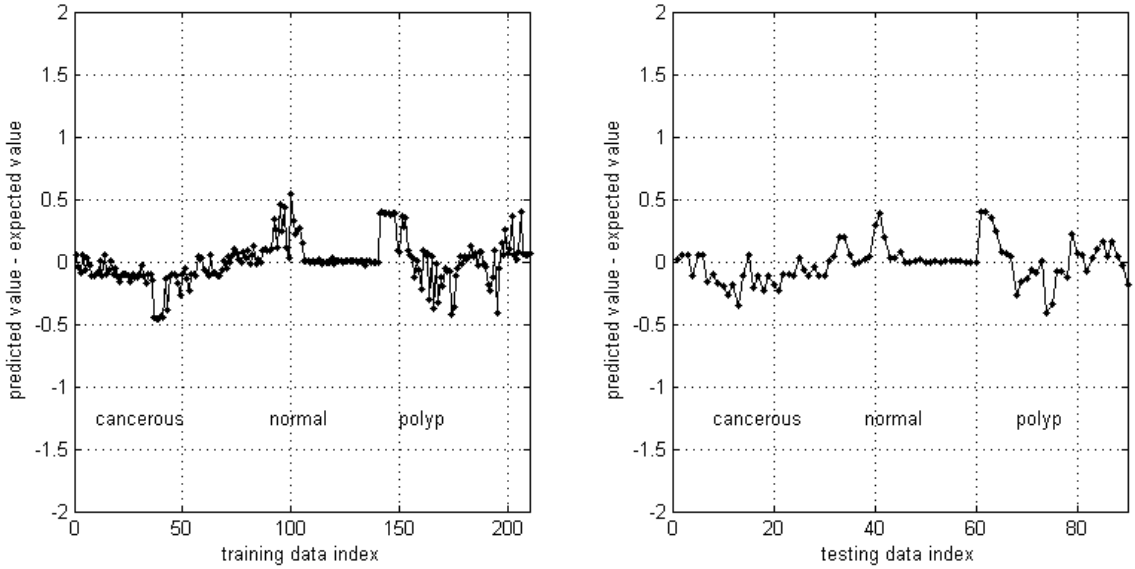


Figure 5.10 Classification Difference of Trained ANFIS using training and testing data sets for Set B features: Mean and Sum average

Table 5.7 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set B features: Mean and Sum average

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.0732	0.5443	0.8865	0.0545	0.5137	0.8945
Predicted Aden. Polyp	0.4321	0.1588	0.3865	0.4483	0.1505	0.3945
Predicted Cancerous	0.9308	0.4557	0.1238	0.9483	0.4863	0.1272

Table 5.8 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set B features: Mean and Sum average with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	62	6	0	28	3	0
Predicted Aden. Polyp	8	50	7	2	24	2
Predicted Cancerous	0	14	63	0	3	28
Percent Accuracy	83.3333%			88.8889%		
Classification Performance Index	0.7419			0.8189		

Set C feature combination [Contrast, entropy, and difference variance]:

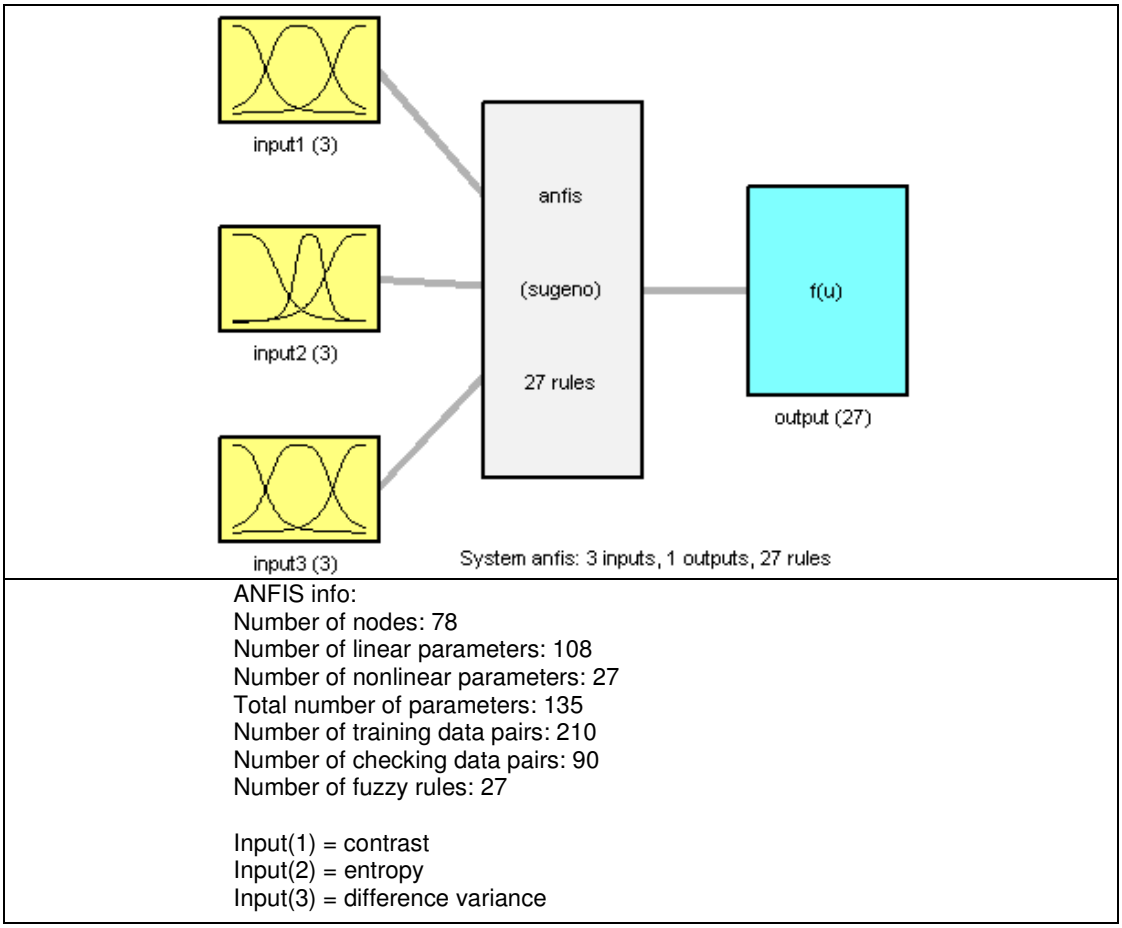


Figure 5.11 ANFIS Structure of Set C features

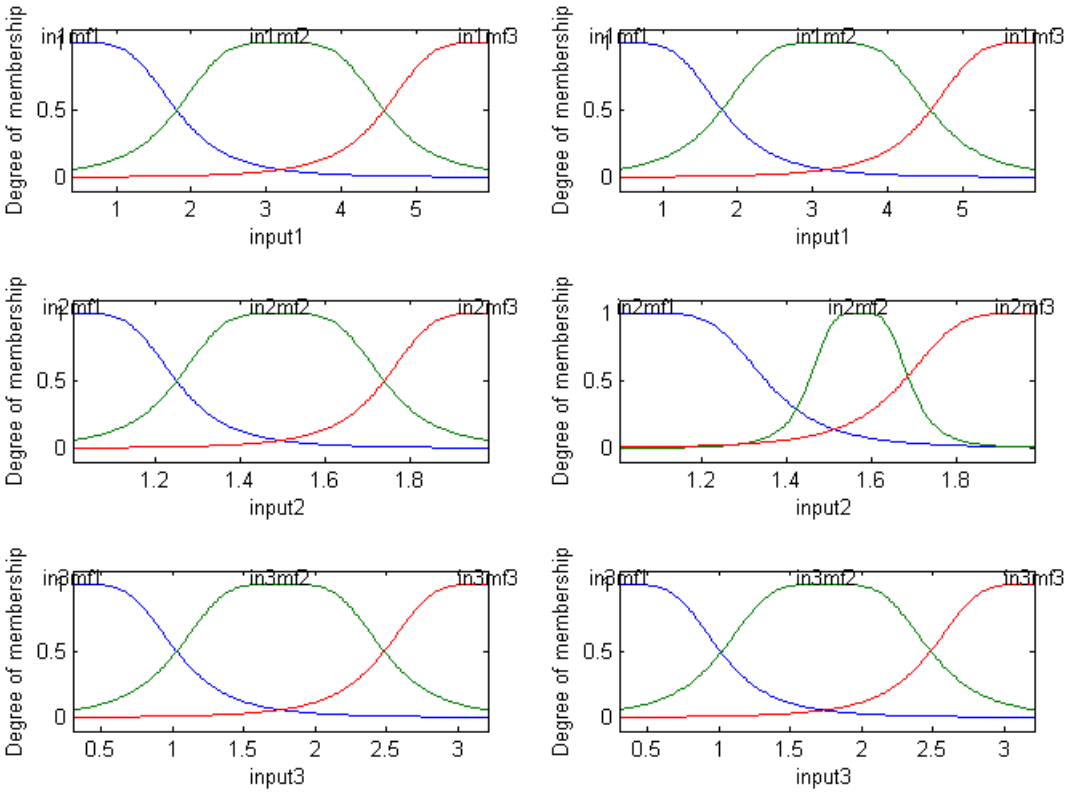


Figure 5.12 ANFIS Membership Functions using Set C features: contrast (input1), entropy (input2), difference variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

The topological arrangement of the input and output variables of the ANFIS using Set C features is shown on Figure 5.11. In Figure 5.11, 3 membership functions were used for each input. Figure 5.12 shows that the membership functions were updated during the training unlike in the cases of Set A and Set B features.

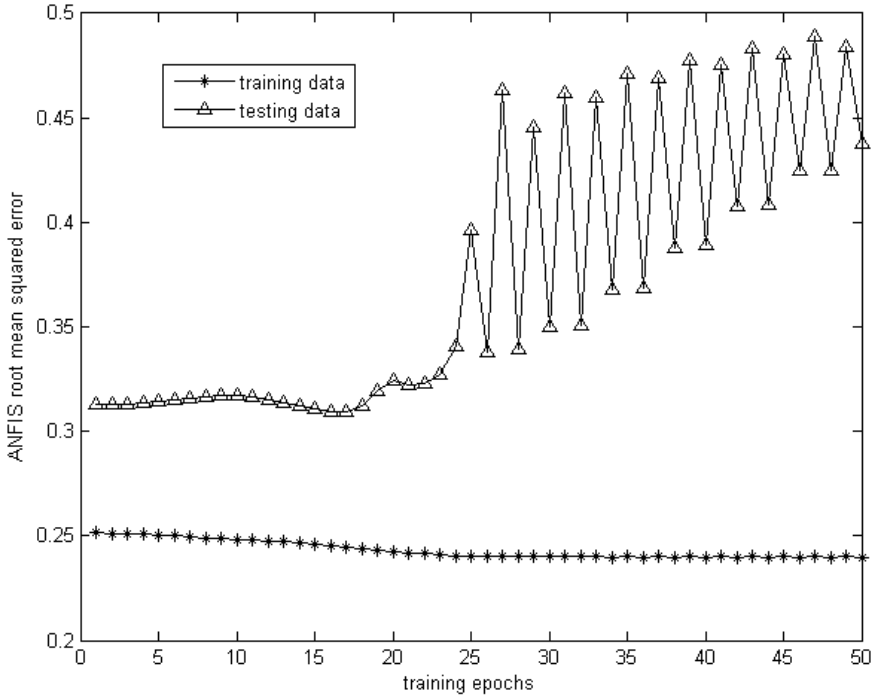


Figure 5.13 ANFIS root mean squared errors during training for Set C features: contrast, entropy, difference variance

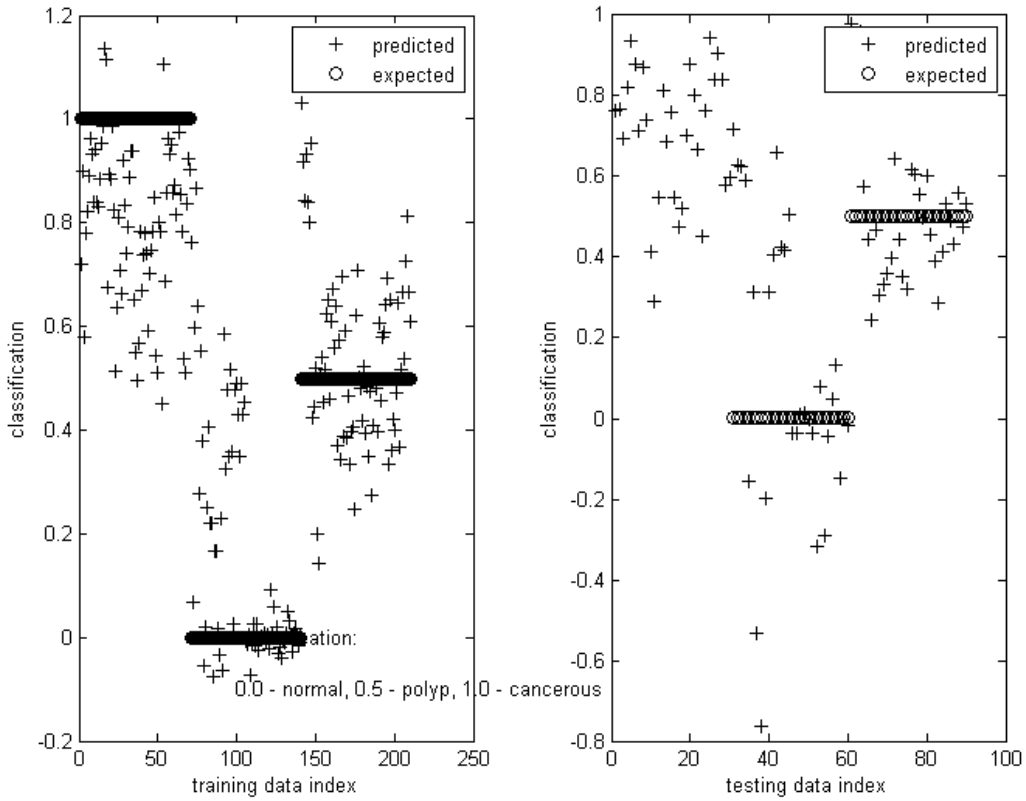


Figure 5.14 Classification performance trained ANFIS using training and testing data sets for Set C features: contrast, entropy, difference variance

The root mean squared errors during the training of the ANFIS are shown on Figure 5.13. On Figure 5.14, the clustering of ANFIS outputs using Set C features is presented for both training and testing images. It can be observed from Figure 5.13, Figure 5.14 and Figure 5.15 that the classifier performed even worse compared to Set A and Set B. The effectiveness of the ANFIS classifier using Set C features is tabulated in Table 5.9 and Table 5.10.

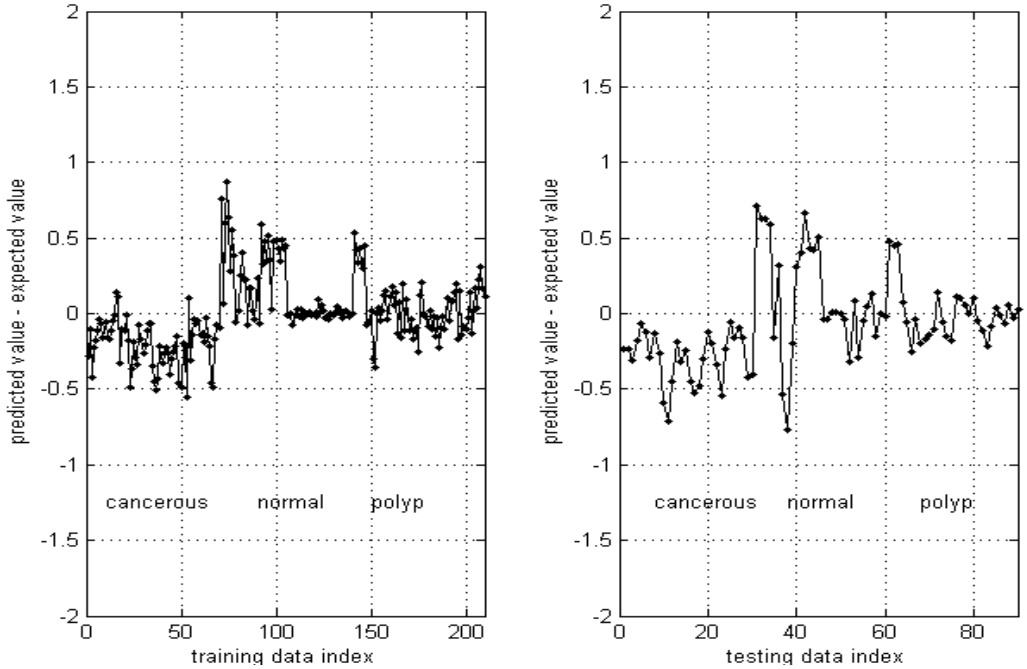


Figure 5.15 Classification Difference of Trained ANFIS using training and testing data sets for Set C features: contrast, entropy, difference variance

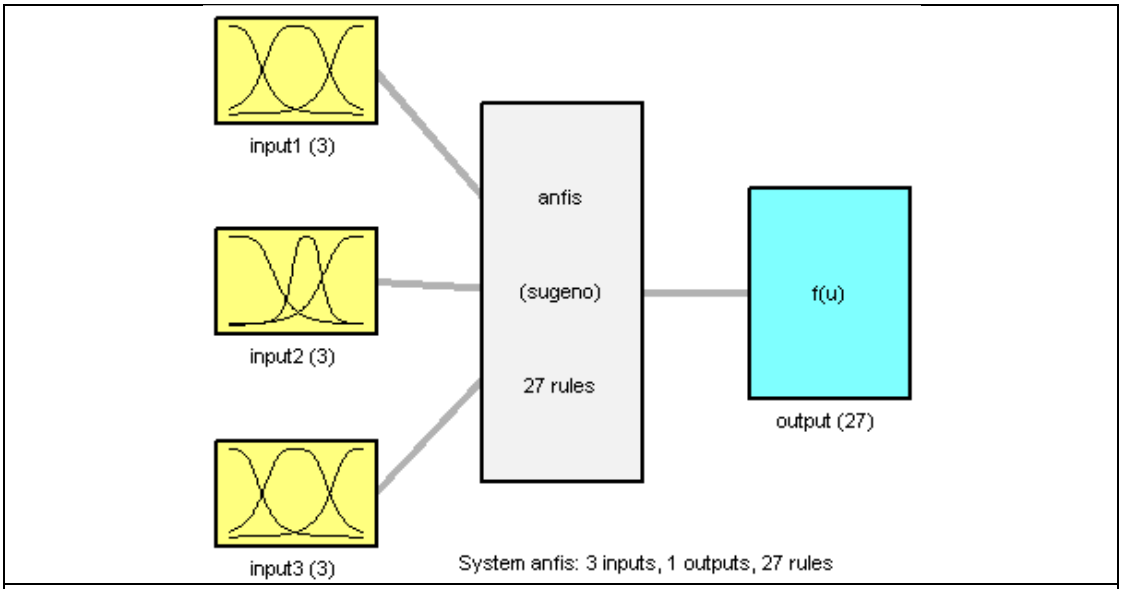
Table 5.9 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set C features: contrast, entropy, difference variance

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.2385	0.7189	1.0639	0.3764	0.6765	0.9429
Predicted Aden. Polyp	0.4883	0.1881	0.3993	0.5843	0.1776	0.309
Predicted Cancerous	1.1161	0.6157	0.283	1.1866	0.6568	0.3904

Table 5.10 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set C features: contrast, entropy, difference variance with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	49	3	0	19	1	0
Predicted Aden. Polyp	19	59	24	11	26	15
Predicted Cancerous	2	8	46	0	3	15
Percent Accuracy	73.3333%			66.6667%		
Classification Performance Index	0.5526			0.4283		

Set D feature combination [Contrast, IDM, and sum variance]:



ANFIS info:
Number of nodes: 78
Number of linear parameters: 108
Number of nonlinear parameters: 27
Total number of parameters: 135
Number of training data pairs: 210
Number of checking data pairs: 90
Number of fuzzy rules: 27

Input(1) = contrast
Input(2) = inverse difference moment or IDM
Input(3) = sum variance

Figure 5.16 ANFIS Structure of Set D features

Figure 5.16 shows the topological arrangement of the input and output variables of the ANFIS using Set D features. It can be noticed from Figure 5.16 that 3 membership functions were used for each input. Figure 5.17 shows that the membership functions were affected during the training process. The root mean squared errors during the training of the ANFIS using Set D are shown on Figure 5.18. On Figure 5.19, the clustering of ANFIS outputs using Set D features is presented for both training and testing images. It can be observed from both Figure 5.19 and Figure 5.20 that the cancerous and polyp cases are the most difficult to classify. The effectiveness of the ANFIS classifier using Set D features is tabulated in Table 5.11 and Table 5.12.

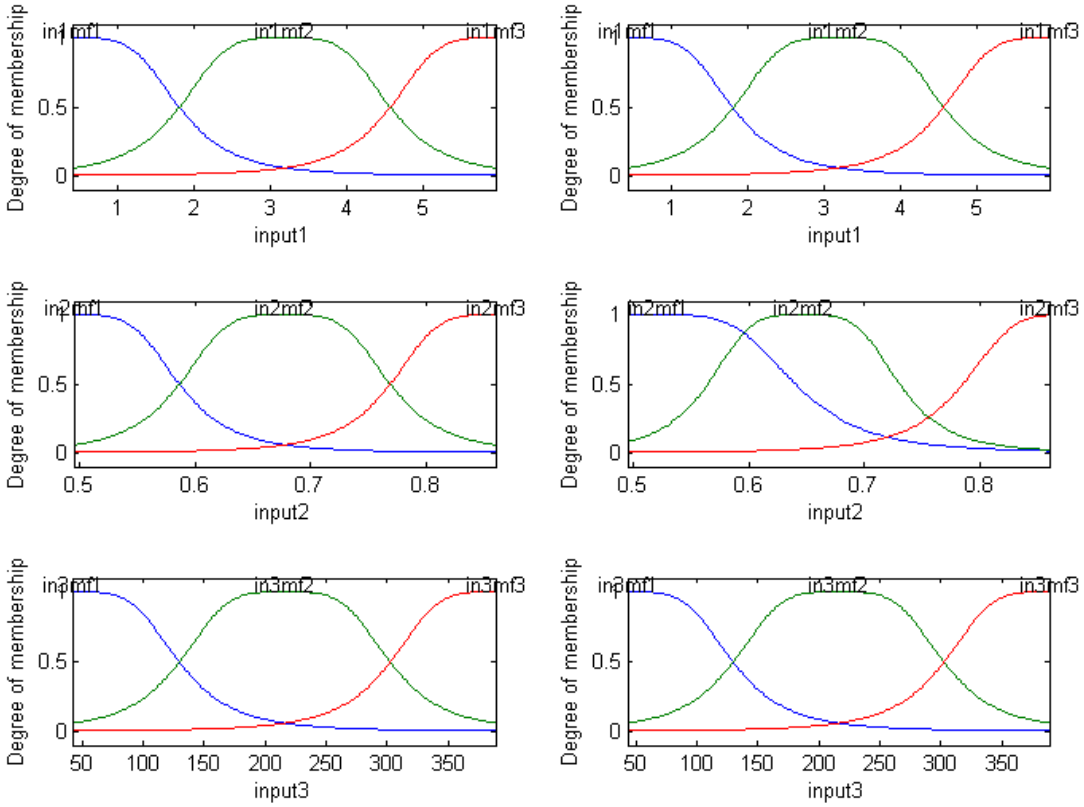


Figure 5.17 ANFIS Membership Functions using Set D features: contrast (input1), inverse difference moment or IDM (input2), sum variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

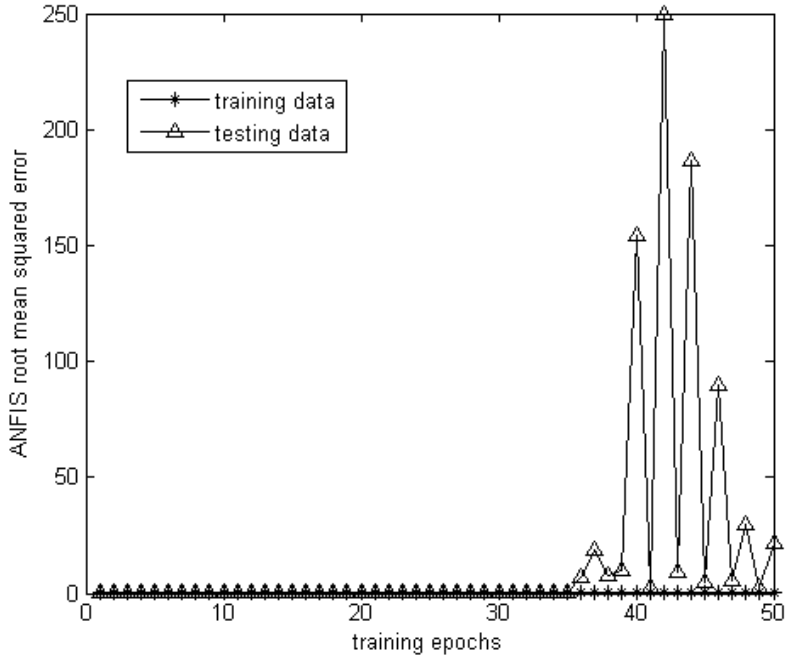


Figure 5.18 ANFIS root mean squared errors during training for Set D features: contrast, inverse difference moment or IDM, sum variance

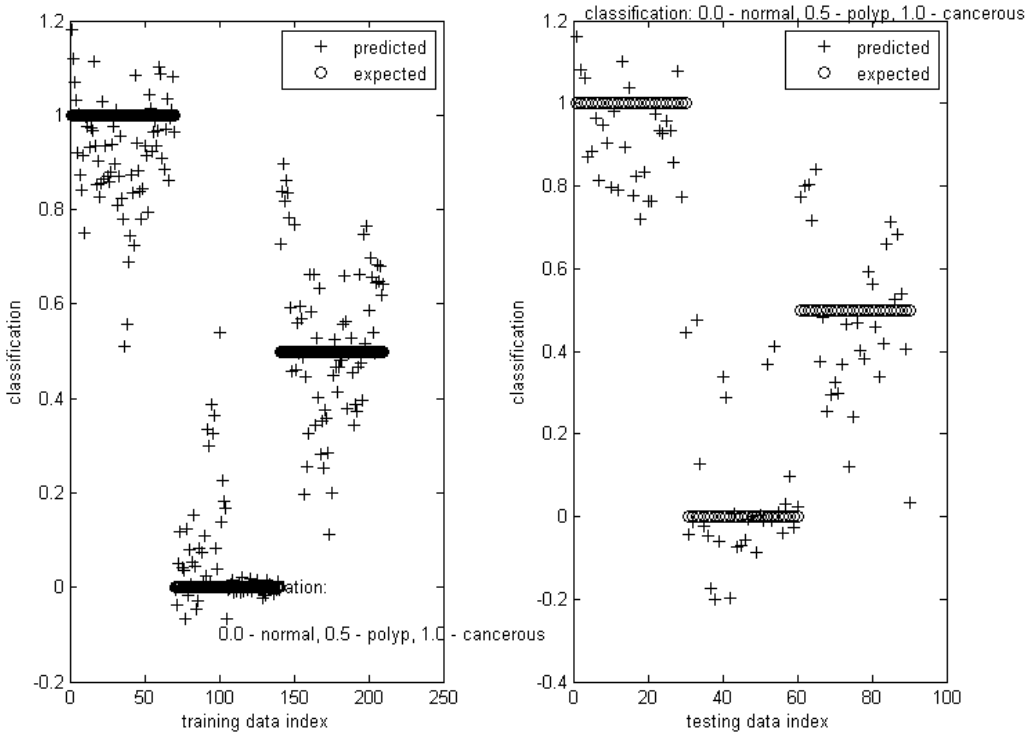


Figure 5.19 Classification performance trained ANFIS using training and testing data sets for Set D features: contrast, inverse difference moment or IDM, sum variance

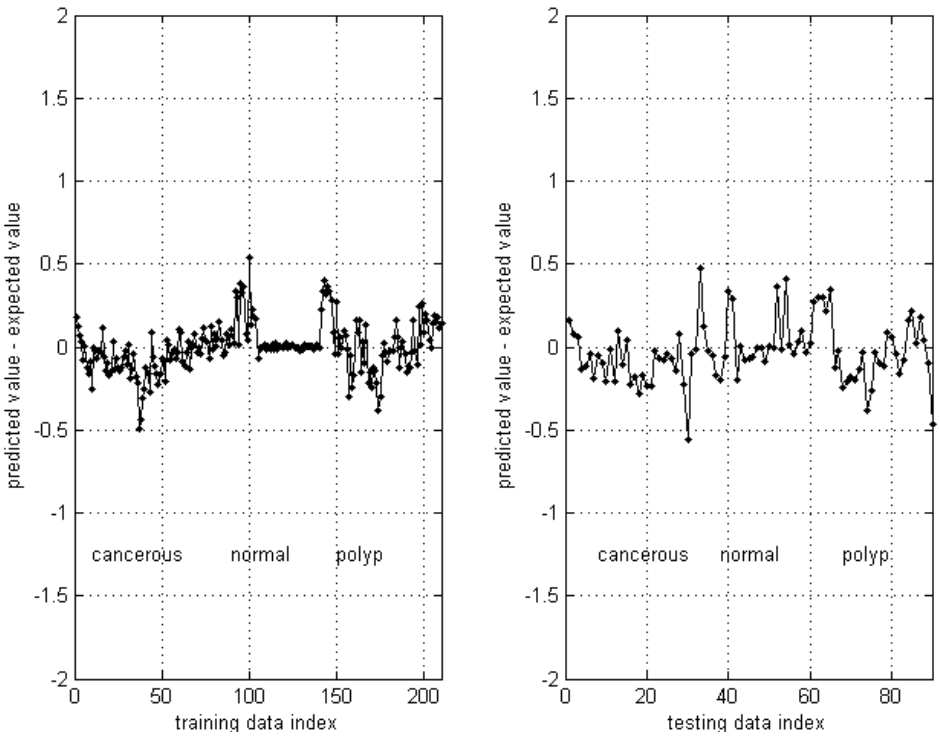


Figure 5.20 Classification Difference of Trained ANFIS using training and testing data sets for Set D features: contrast, inverse difference moment or IDM, sum variance

Table 5.11 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set D features: contrast, inverse difference moment or IDM, sum variance

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.089	0.7037	1.2232	0.1478	0.637	1.1928
Predicted Aden. Polyp	0.595	0.1856	0.5565	0.6209	0.2257	0.531
Predicted Cancerous	1.2602	0.6297	0.1489	1.2875	0.6963	0.1865

Table 5.12 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set D features: contrast, inverse difference moment or IDM, sum variance with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	64	3	0	25	3	0
Predicted Aden. Polyp	6	59	5	5	23	2
Predicted Cancerous	0	8	65	0	4	28
Percent Accuracy	89.5238%			84.4444%		
Classification Performance Index	0.8381			0.7661		

Set E feature combination [Sum average and difference entropy]:

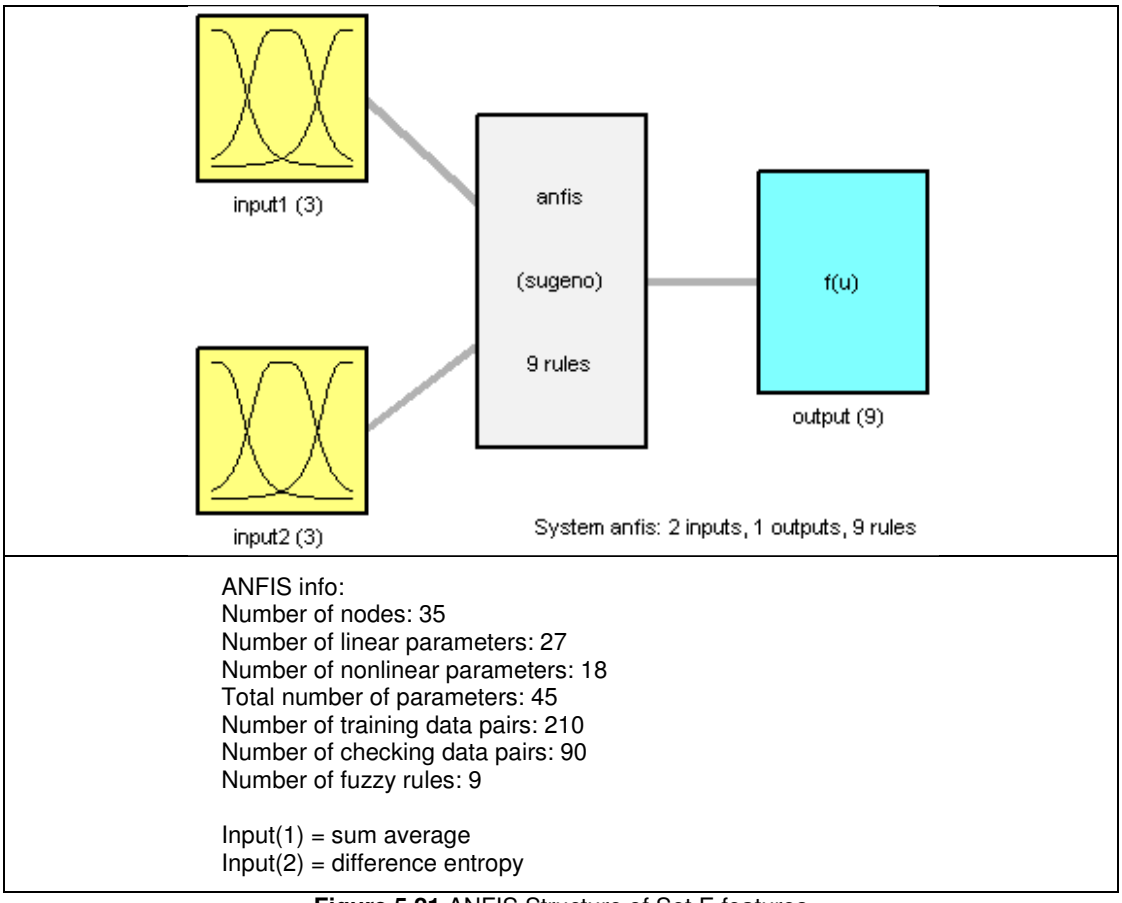


Figure 5.21 ANFIS Structure of Set E features

Figure 5.21 shows the topological arrangement of the input and output variables of the ANFIS using Set E features. It can be noticed from Figure 5.21 that 2 membership functions were used for each input. Figure 5.22 shows that the membership functions remained unchanged during the training process.

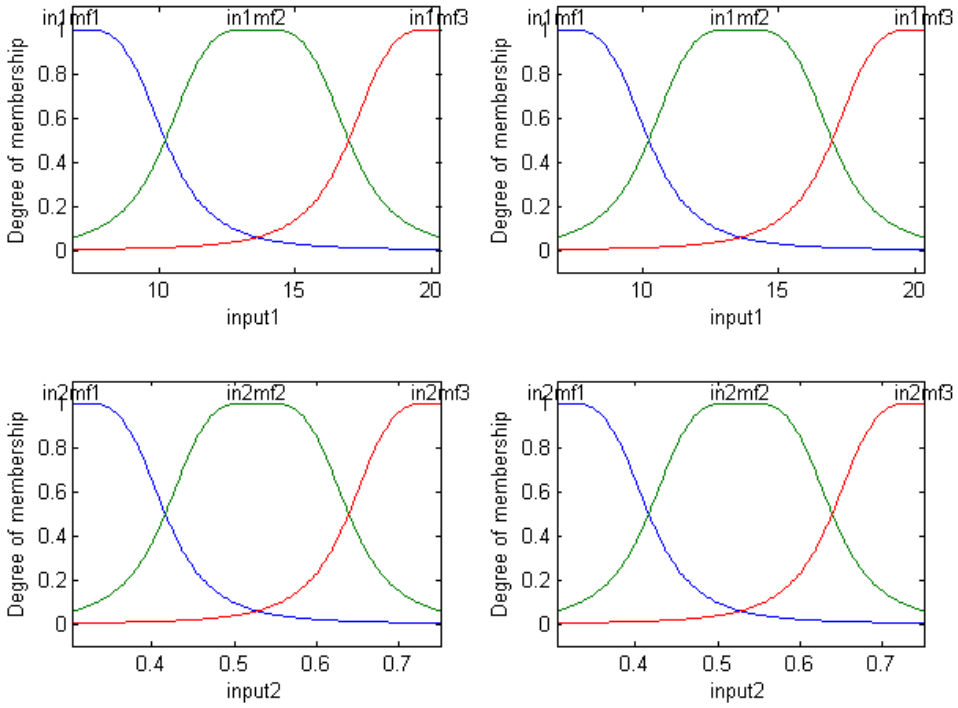


Figure 5.22 ANFIS Membership Functions using Set E features: sum average (input1), difference entropy (input2). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

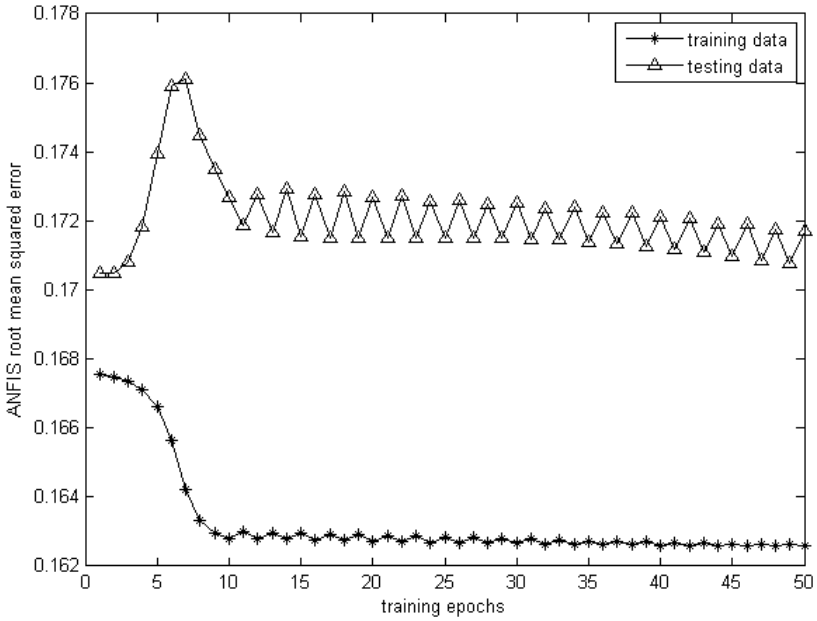


Figure 5.23 ANFIS root mean squared errors during training for Set E features: sum average, difference entropy

The root mean squared errors during the training of the ANFIS are shown on Figure 5.23. On Figure 5.24, the clustering of ANFIS outputs using Set E features is presented for both training and testing images. It can be observed from both Figure 5.24 and Figure 5.25 that the cancerous and polyp cases are the most difficult to classify. The effectiveness of the ANFIS classifier using Set E features is tabulated in Table 5.13 and Table 5.14.

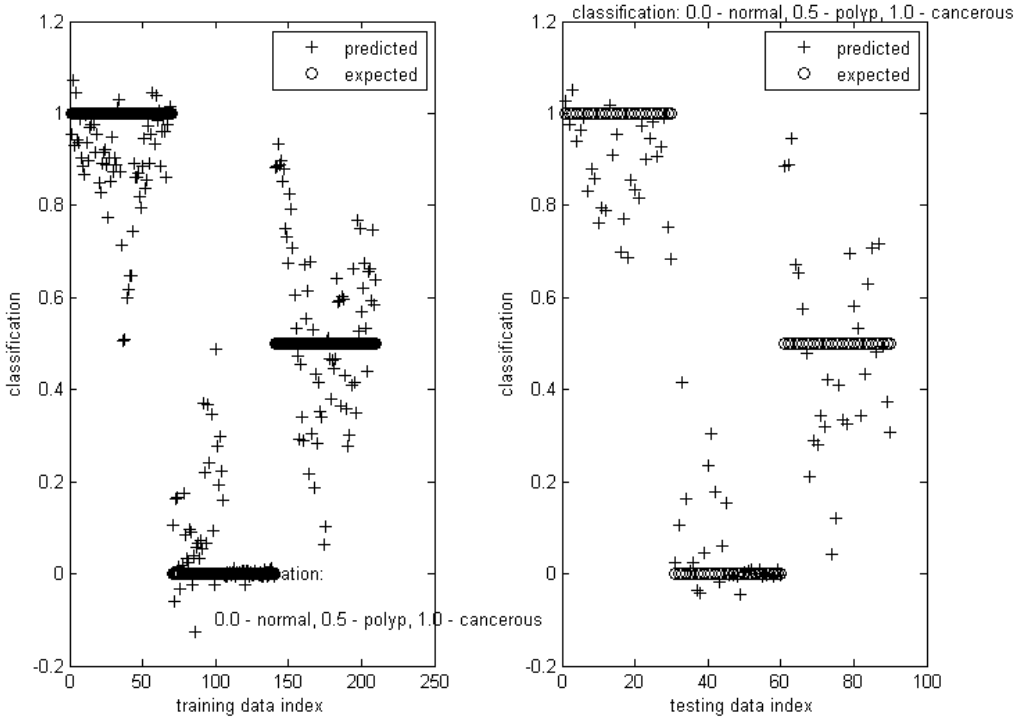


Figure 5.24 Classification performance trained ANFIS using training and testing data sets for E features: sum average, difference entropy

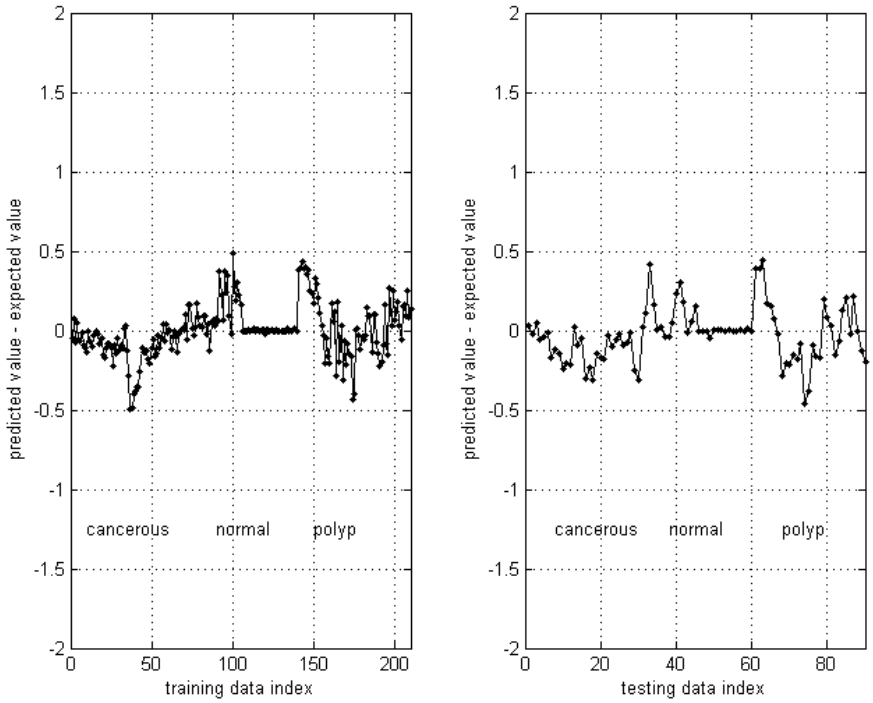


Figure 5.25 Classification Difference of Trained ANFIS using training and testing data sets for Set E features: sum average, difference entropy

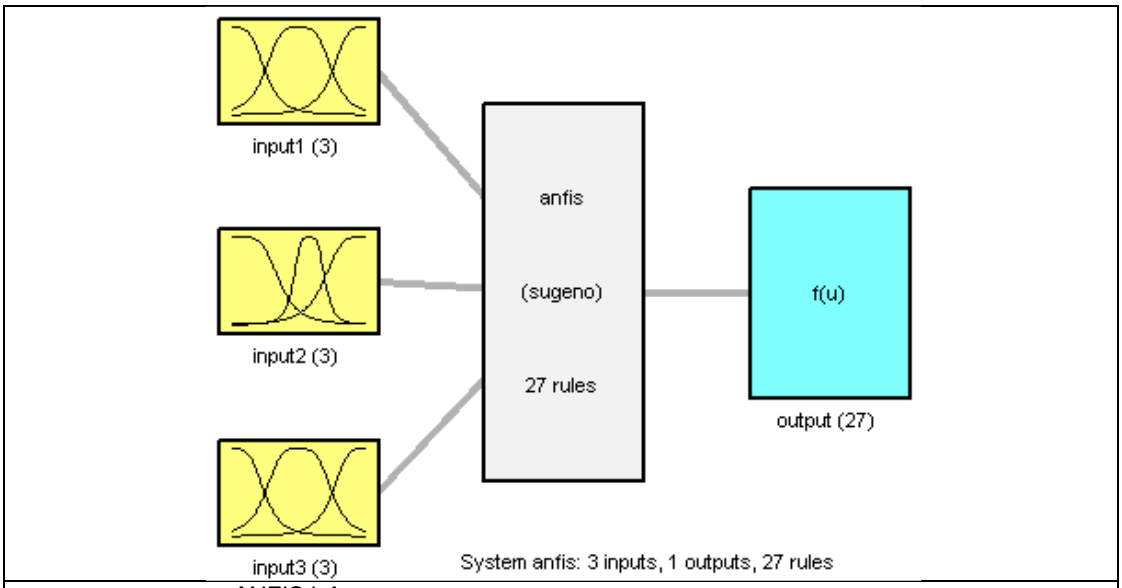
Table 5.13 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set E features: sum average, difference entropy

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.0728	0.5434	0.894	0.0641	0.4829	0.882
Predicted Aden. Polyp	0.4374	0.1688	0.394	0.4471	0.1824	0.382
Predicted Cancerous	0.9374	0.4566	0.1136	0.9471	0.5171	0.1245

Table 5.14 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set E features: sum average, difference entropy with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	64	4	0	28	3	0
Predicted Aden. Polyp	6	56	8	2	24	3
Predicted Cancerous	0	10	62	0	3	27
Percent Accuracy	86.6667%			87.7778%		
Classification Performance Index	0.7852			0.7944		

Set F feature combination [Contrast, inverse difference moment or IDM, and difference variance]:



ANFIS info:
Number of nodes: 78
Number of linear parameters: 108
Number of nonlinear parameters: 27
Total number of parameters: 135
Number of training data pairs: 210
Number of checking data pairs: 90
Number of fuzzy rules: 27

Input(1) = contrast
Input(2) = inverse difference moment or IDM
Input(3) = difference variance

Figure 5.26 ANFIS Structure of Set F features

Figure 5.26 shows the topological arrangement of the input and output variables of the ANFIS using Set F features. It can be noticed from Figure 5.26 that 3 membership functions were used for each input. Figure 5.27 shows that the membership functions were affected during the training process. The root mean squared errors during the training of the ANFIS using Set F are shown on Figure 5.28. On Figure 5.29, the clustering of ANFIS outputs using Set F features is presented for both training and testing images. It can be observed from both Figure 5.29 and Figure 5.30 that the

cancerous and polyp cases are the most difficult to classify. The effectiveness of the ANFIS classifier using Set F features is tabulated in Table 5.15 and Table 5.16.

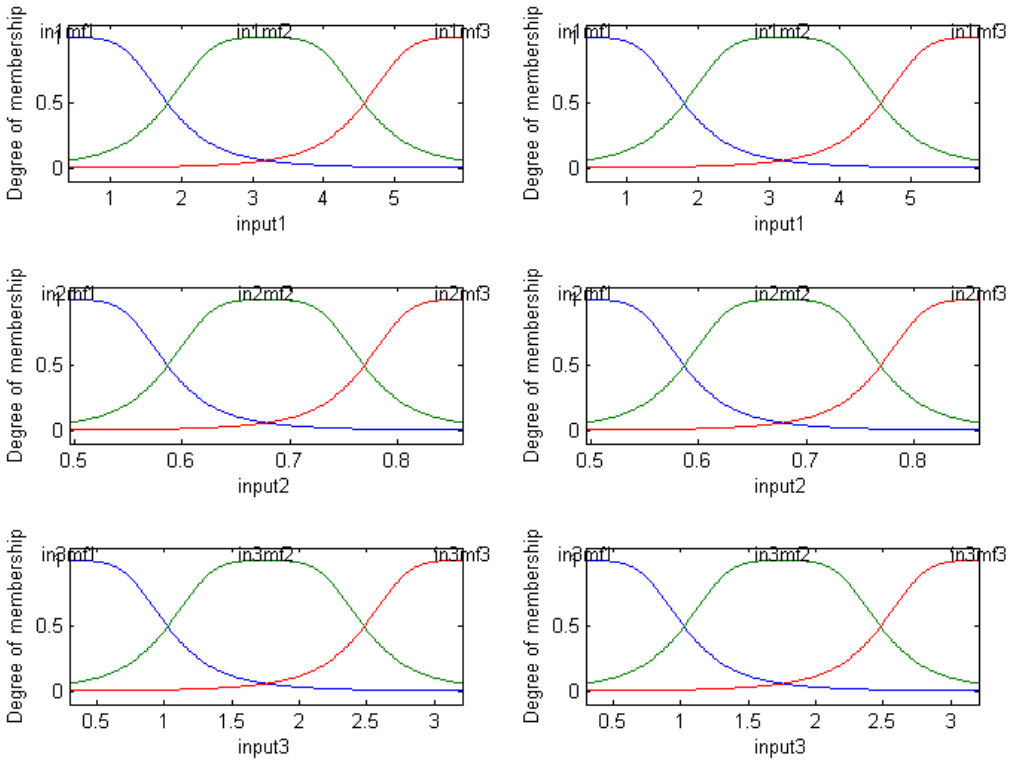


Figure 5.27 ANFIS Membership Functions using Set F features: contrast (input1), inverse difference moment or IDM (input2), difference variance (input3). Left side plots are refer to 'before training' while the right side plots refer to 'after training'.

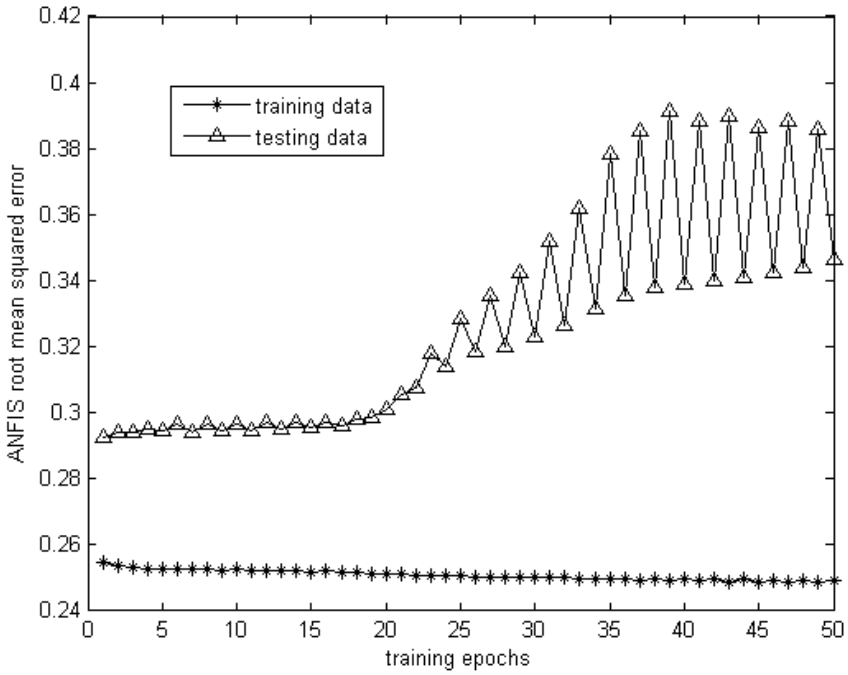


Figure 5.28 ANFIS root mean squared errors during training for Set F features: contrast, inverse difference moment or IDM, difference variance

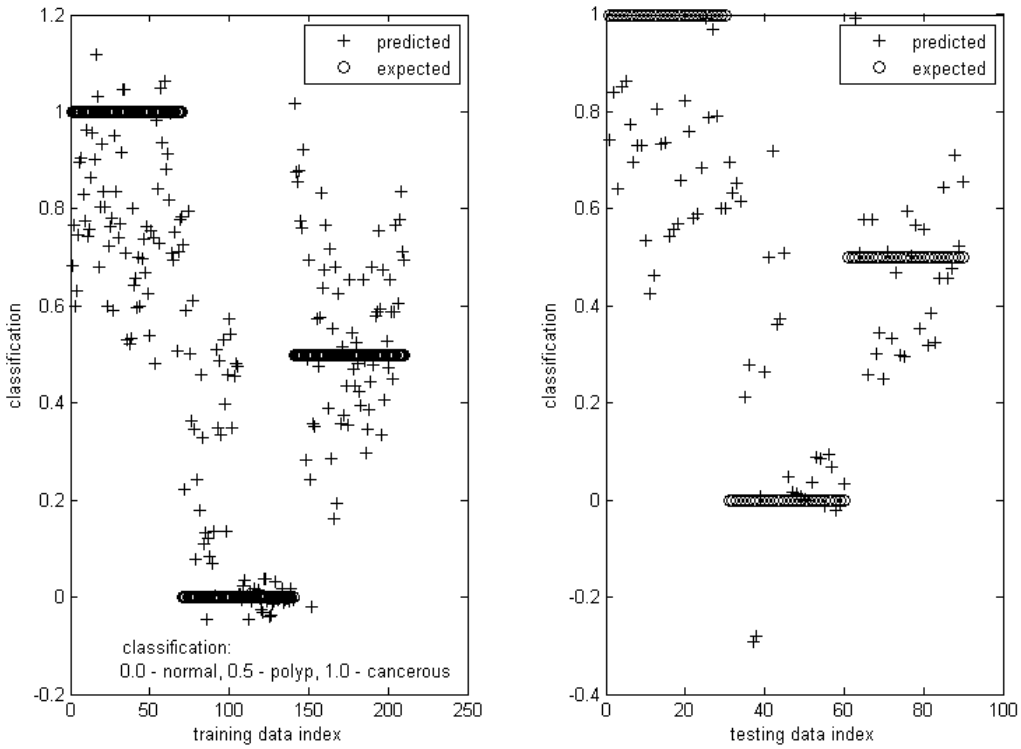


Figure 5.29 Classification performance trained ANFIS using training and testing data sets for Set F features: contrast, inverse difference moment or IDM, difference variance

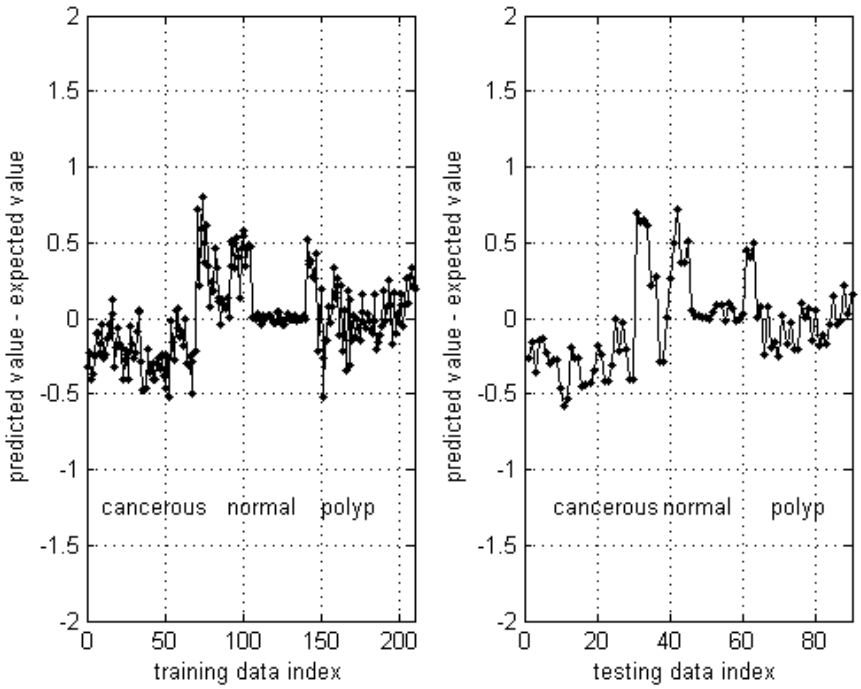


Figure 5.30 Classification Difference of Trained ANFIS using training and testing data sets for Set F features: contrast, inverse difference moment or IDM, difference variance

Table 5.15 Mean Relative Difference Confusion Matrix (MRDCM) for training and testing data sets using Set F features: contrast, inverse difference moment or IDM, difference variance

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	0.2431	0.733	1.0368	0.3083	0.6713	0.9369
Predicted Aden. Polyp	0.4692	0.217	0.3709	0.487	0.1981	0.2801
Predicted Cancerous	1.1024	0.6017	0.3101	1.08	0.6621	0.3964

Table 5.16 Confusion matrix, percent accuracy, and classification performance index (CPI) for training and testing data sets using Set F features: contrast, inverse difference moment or IDM, difference variance with threshold values of 0.25 and 0.75

--	Training Data Set			Testing Data Set		
	Expected Normal	Expected Aden. Polyp	Expected Cancerous	Expected Normal	Expected Aden. Polyp	Expected Cancerous
Predicted Normal	48	4	0	19	1	0
Predicted Aden. Polyp	21	53	31	11	26	19
Predicted Cancerous	1	13	39	0	3	11
Percent Accuracy	66.6667%			62.2222%		
Classification Performance Index	0.436			0.3306		

5.3 Image Classification by Human Pathologists

Classification of histopathological images has always been exclusively within the scope of the domain of human pathologists. It is therefore logical to hold the classification performance of human pathologists as a benchmark in developing an automatic histopathological image classifier. With this idea in mind, some practicing pathologists in Manila were requested to participate in a survey or test of classifying 15 images that were derived at random from the test image set. The random selection of images from the test image set was reached in such a way that each of the classes would be equally represented. The 15 test images therefore contained 5 images for each of the normal, adenomatous polyp, and cancerous cases. The number of images used in the survey test was small compared to the total of 90 images in the test image set used in the entire study. This number was carefully chosen in order for the pathologists not to view the survey test as a burden on their part since most if not all of them are busy people. It was felt that more than 15 images might be asking too much from them considering that the survey test could be viewed as a form of scrutiny of their abilities.

Ten experienced pathologists were invited to participate but only six of them agreed to take-up the challenge.

The test was conducted by presenting each pathologist with 5 pages of a survey form with 3 monochromatic images printed on each page. Each image was printed with dimensions of approximately $3\,{}^{1}/_{8}$ inches by $4\,{}^{3}/_{16}$ inches on an A4-sized bond paper. The task given to each pathologist was simply to classify each image as normal, adenomatous polyp, or cancerous case. To eliminate obvious trending, the images were arranged in a random fashion throughout the entire survey form. Part of the conditions that were promised to the pathologists was anonymity on their part and therefore in this report, they are identified as pathologists A, B, C, D, E, and F. Table 5.17 shows the results of the test survey with corresponding data on the number of years of experience of each pathologist.

Table 5.17 Results of the test survey conducted on 6 pathologists using 15 monochromatic colonic images selected randomly from the test image set. CPI stands for classification performance index.

Pathologist ID	Years of Experience	Confusion Matrix			accuracy	CPI
Pathologist A	30	5	0	0	86.667 %	0.706667
		0	5	2		
		0	0	3		
Pathologist B	30	5	0	0	93.333 %	0.853333
		0	5	1		
		0	0	4		
Pathologist C	25	4	3	0	66.667 %	0.396667
		1	2	1		
		0	0	4		
Pathologist D	14	5	3	1	73.333 %	0.453333
		0	2	0		
		0	0	4		
Pathologist E	5	5	2	0	80 %	0.66
		0	2	0		
		0	1	5		
Pathologist F	11	5	3	0	66.667 %	0.326667
		0	2	2		
		0	0	3		

It can be observed from Table 5.17 that the pathologists handled the normal cases well. However, the story with the non-normal images was different. Most of them had

misclassifications on the adenomatous polyp cases while all of them made mistakes in the cancerous cases. This indicates that non-normal images are much harder to classify. Table 5.17 also suggests that pathologists with 30 years of experience might have higher classification accuracy compared to pathologists with less experience. Figure 5.31 shows the comparison of the classification performances between the pathologists and the ANFIS algorithms using the different texture property sets. It is clear that pathologist B performed best while the ANFIS algorithms using sets A, B, and E also did fairly well. The plots in Figure 5.31 also show that the CPI parameter emphasises misclassification which is why it tends to exhibit proportionally lower scores for classifiers with more mistakes.

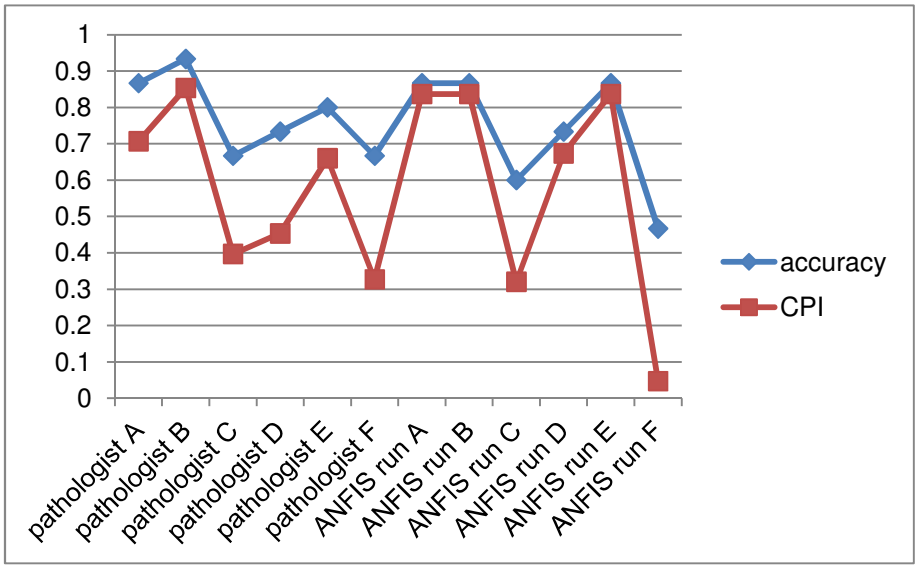


Figure 5.31 Comparison between the classification accuracy and classification performance index (CPI) of the pathologists and ANFIS algorithms using texture properties.

Figure 5.32 illustrates further why the CPI parameter is a better parameter than the classification accuracy. One can notice that the information given by the bar plots in Figure 5.32 indicate contradiction. The accuracy parameter informs that, on average, the pathologists performed better while the opposite is given by the CPI parameter.

This is understandable since the accuracy parameter does not take into account the gravity of mistakes committed by a classifier in question. The average CPI for the ANFIS algorithm is higher because it performed better in the upper off-diagonal part of the confusion matrices. This was never detected by the accuracy parameter.

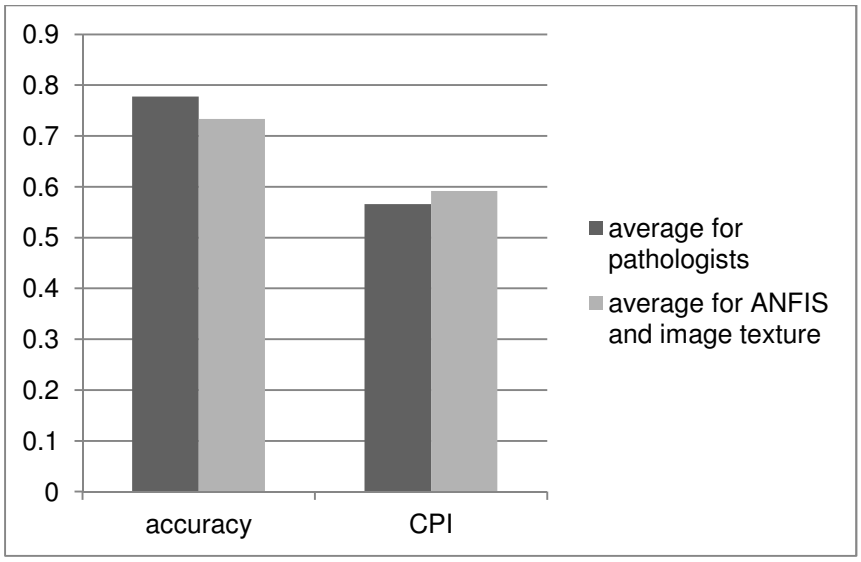


Figure 5.32 Average accuracy and CPI values for the pathologists and the ANFIS algorithm with different feature sets.

5.4 Summary of the Image Classification Implementation

This chapter has presented the results of testing the classification performances of the different feature combinations suggested in Chapter 4 with the use of ANFIS as classifier and the training images as training set. A number of performance indicators such as plots and tables were generated and presented to show the success and failures of ANFIS given different feature combinations. The plots showing the behaviour of the ANFIS root mean squared errors during training illustrated how the classifier coped with each feature combination. Some feature combinations obviously

made the classification of images more 'difficult' compared to other feature combinations. Two performance measures were also introduced in this chapter: the Mean Relative Difference Confusion Matrix (MRDCM) and the Classification Performance Index or CPI. The purpose of MRDCM is to allow clinicians or pathologists to make use of the real number output of the ANFIS classifier and thereby avoid the use of threshold values to characterise an image. The CPI on the other hand is considered here as a better alternative to the percent accuracy parameter when expressing the classification quality reflected by a confusion matrix. The CPI utilises a set of numbers called factor matrix that collectively imposes a kind of penalty to elements in the confusion matrix that represent bad classification performance. It was pointed out that one of the disadvantages of using the percent accuracy is that it does not distinguish between bad and worse misclassifications. An example of this is misclassification of cancerous into adenomatous polyp compared to misclassification of cancerous into normal. Unlike the percent accuracy parameter, the CPI puts more 'penalty' on the latter case of misclassification.

Figure 5.31 summarises and compares the CPI and the percent accuracy (in decimal) of the classifier given the different feature combinations. It should be noted that the CPI parameter does not have a percent version since it does not account for a percentage of anything. The CPI is merely a rating from 0.0 to 1.0. The trend between the CPI and accuracy is clear. Both are reflections of the relative performances of the different feature combinations. To view the results from a different viewpoint, Table 5.17 aggregates together the normalised confusion matrices for the different feature combinations. One common achievement among the feature combinations is the fact that none of them misclassified any cancerous case into normal and vice versa.

Together with Table 5.18, Figure 5.33 presents a clear picture of the comparison of classification performances of the different feature combinations used in this research

using the test image set and ANFIS as classifier. From these two presentations, it is clear that Set A, Set B and Set E feature sets gave excellent results.

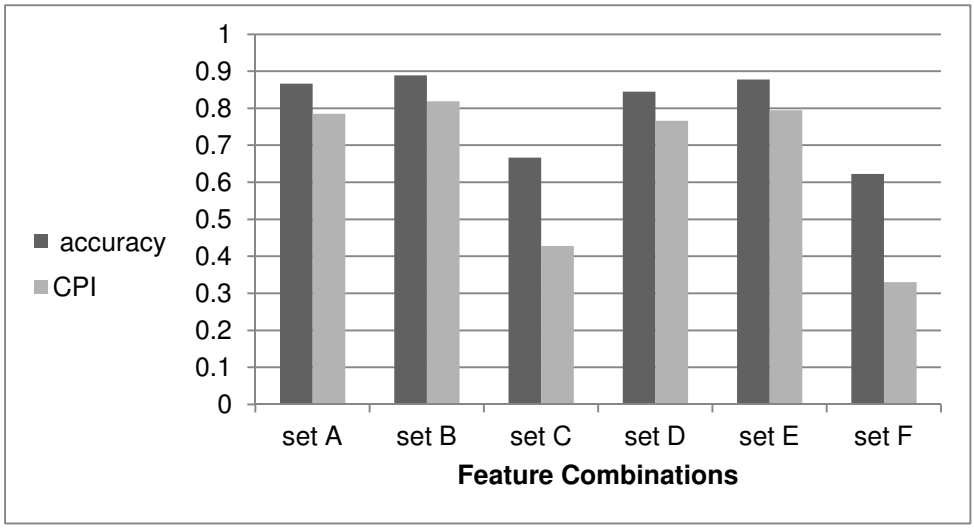


Figure 5.33 Comparison of classification performances of the different feature combinations used in this research using the test image set and ANFIS as classifier. The CPI is the classification performance index while the accuracy is the decimal version of the percent accuracy.

Table 5.18 Summary of the normalized confusion matrices of the different feature combinations using the test image set and ANFIS as classifier. The columns represent the expected classifications while the rows are the predicted classifications. From left to right and from top to bottom, the classes are normal, adenomatous polyp, and cancerous.

Set A				Set D		
90	13.33333	0		83.33333	10	0
10	76.66667	6.666667		16.66667	76.66667	6.666667
0	10	93.33333		0	13.33333	93.33333
Set B				Set E		
93.33333	10	0		93.33333	10	0
6.666667	80	6.666667		6.666667	80	10
0	10	93.33333		0	10	90
Set C				Set F		
63.33333	3.333333	0		63.33333	3.333333	0
36.66667	86.66667	50		36.66667	86.66667	63.33333
0	10	50		0	10	36.66667

Finally, the results of a survey test conducted on the classification abilities of some human pathologists were presented. This step was necessary because establishing a

benchmark for classification performance provides a basis for evaluating any automatic classifier under development. Table 5.17 suggests that the classification of normal images is not difficult for pathologists while the adenomatous polyp and cancerous cases are more difficult to classify. Figure 5.32 points out that the shortcomings of classifier systems are better accounted for by the use of the CPI parameter. Although the pathologists were presented with a random selection of images from the three categories, it must be emphasised that these were of the same level of complexity as the ones analysed through the algorithms developed in this study. However, since the pathologists only considered 15 images to classify, this presents a limitation to the study and may not be enough to make valid general conclusions or comparison between the performances of the algorithms and the pathologists.

5.5 Comparison of Results with Previous Studies

The usual percentage accuracy of classifiers seems to be close to around 90%. Esgiar *et al.* (1999) used 44 normal and 58 cancerous images subjected to linear discriminant analysis. Using fractal analysis together with entropy and correlation textural features, a 94% accuracy was reported to have been achieved. In 2001, Atlamazoglou *et al.* also used GLCM to extract features from 70 fluorescence images of colonic tissue sections to achieve 95% classification accuracy. The features that were used were inverse difference moment, correlation, the f_{12} and f_{13} measures of correlation with a Mahalanobis distance linear discriminant classifier. Tjoa and Krishnan (2002) used texture properties from 66 coloured microscopic images to achieve a 92.42% accuracy on back-propagation neural network, while only reaching a 83.33% accuracy using unsupervised networks. In 2004, Nwoye et al. reported a classification accuracy of 96.4% from 116 cancerous and 88 normal colon cell images.

A fuzzy-neural network combined with a clustering algorithm was proposed in that study where fractal dimension techniques and textural features were used. The textural features that were used were entropy, correlation, inverse difference moment, and angular second moment. Filippas et al. (2003a) was able to achieve 100% and 91% accuracy in some cases by implementing genetic algorithms on a Parallel Virtual Machine or PVM. There were three feature groups that were used: features from image histogram, grey-level difference statistic and GLCM. The GLCM features used were mean, variance, contrast, entropy and angular second moment. Filippas et al. (2003b) later used BPNN to achieve 87.5% accuracy. More recently, Fiscor et al. (2008) classified between 24 normal mucosa, 11 aspecific colitis, 25 ulcerative colitis and 9 cases of Crohn's disease. The overall classification accuracy was 88% using leave-one-out discriminant analysis.

In this study, Figure 5.33 shows that feature sets A, B, D and E all yield accuracies close to what others have been obtaining – around 90%. However, this is not to say that the figures on classification accuracy from different studies can be precisely compared. Comparison makes sense in this case only if the focus is on the trend of the values mentioned. Part of this trend is the fact that the entropy property appears to be a reliable feature since it is used frequently.

Chapter 6 - CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

During the research topic proposal stage, the original idea was to devise a single algorithm that combines fuzzy logic (FL), neural networks (NN), and genetic algorithms (GA) paradigms to automatically classify colonic histopathological images. It was later decided to divide the research problem into two parts: a feature selection part and a classification part. This time, instead of considering a single unified hybrid algorithm for the whole study, two levels of algorithms were conceptualised. This decision proved to be an important one because it allowed for the application of a strategy based on divide-and-conquer. As a result, this research has put forward a number of ideas and findings. One element that did not change throughout the study was the use of texture properties derived from a grey level co-occurrence matrix or GLCM.

The ratio of variances based on the modified criterion from Multiple Discriminant Analysis (MDA) as used also by Boland *et al.* (1998) and Altlamazoglou *et al.* (2001) showed excellent results. The procedure was simple and straightforward. It was a tool that rated each individual feature according to its clustering effectiveness. Based on Figures 4.2 to Figure 4.10 (except Figure 4.9), the mean, sum average, and sum variance were shown to be the most discriminating textural features. It was also observed that varying the image quantisation levels from 32, 24, 16 to 8 and changing the image size had no effect on the textural properties. However, it was evident that histogram equalisation, which is an image processing operation that reorders the pixel values, did affect the texture of an image and therefore should be avoided prior to extraction of textural properties.

The use of GA-KSOM for the feature selection showed tremendous promise. An advantage of this approach in feature selection is the fact that the information on the classes of the data from a teacher was not needed because of the self-organising nature of the KSOM in the fitness function. The map quality proved to be an excellent parameter in evaluating candidate feature sets. The most crucial part of the implementation of the GA-KSOM feature selection process was the creation of a fitness function that allowed the GA to select "good" feature sets comprised of not "too many" features. The introduction of the penalty function was effective in guiding the GA search to find superior feature sets having a small number of features. In this study, one of the implicit goals of the feature selection process was the discovery of the minimum number of relevant features that would optimise classifier performance. Results of the application of the GA-KSOM feature selection suggested that a clear global optimum was not achieved. This meant that the problem might be multi-modal or the fitness function that was used still required further refinement. Further investigation of the fitness function could yield a more uniform output thus proving that a global optimum indeed exists. Based on this method and by testing with an ANFIS classifier, among the optimum feature combinations were:

Set D - Contrast, inverse difference moment or IDM, and sum variance

Set E - Sum average and difference entropy.

ANFIS was shown to be well-suited for the problem in this study since its output, being a real number from 0.0 to 1.0, could truly represent the classification spectrum that is evident with the way humans characterise colonic images. For the same reason, the single output limitation of ANFIS did not pose a problem despite of the 3 output classes. The relationship between the classes themselves became the solution as to how ANFIS could be utilised as a classifier. However, it was observed that the performance of the ANFIS classifier depended on the chosen feature set. This was to be expected since ANFIS is simply a classifier and not a feature selector by design.

ANFIS is not the only method of combining neural networks with fuzzy logic. There are also a number of suggested schemes of combining fuzzy logic with genetic algorithms and neural networks with genetic algorithms. It is recommended to examine other architectures and compare the results with the findings in this study.

The Mean Relative Difference Confusion Matrix or MRDCM was a natural extension of the ANFIS classifier since the conventional confusion matrix could not be used with the ANFIS output without resorting to selecting threshold values for the 3 classes. This novel confusion matrix supported the idea of providing the human pathologist or the user with more information by pointing out the state of the image in question relative to extreme cases in the normal-to-cancerous spectrum. Thinking in terms of numbers in the classification spectrum might promote more objectivity on the part of the pathologist. The classifier algorithm/s developed in this study was never meant to replace human experts but rather be used as effective supporting tools.

Sometimes it is necessary to express the performance of a classifier through a single number such as the percent accuracy computed from the conventional confusion matrix. A Classification Performance Index, or CPI, was devised in this study as a better parameter than percent accuracy and is as simple to use. The advantage of using CPI is in its ability to account for the successes and severity of failures of a classifier. The CPI parameter achieved this through the use of a novel matrix known as *factor matrix*, also devised in this study.

Figure 5.31 confirmed the so-called inter-observational variation that sometimes exists among pathologists who analyse the same set of histologic images. This might be due to variation in the abilities of pathologists and/or to the fact that misclassifications are not truly mistakes but more like variations in “professional” judgements. Based on the results of the classification test performed on pathologists in this study, the automatic classifiers that were developed here already performed comparatively well. However if one is to assume that the classes provided in the

training and testing data sets were without mistakes, then there is still room for improvement of the algorithms in this study. The pathologists who participated in the survey were only given 15 images to classify. This small number makes it difficult to generalise the result of the comparison between the algorithm performance and that of a human expert.

As for recommendations for future work, one idea is that there are other excellent formulations of neuro-fuzzy architectures that can be considered. In particular, the use of fuzzy neurons can be explored. Another idea that can be developed is the use of online training mode for the classifier. This would enable the classifier to act as a human being where experience can generally improve performance as more and more samples get processed. It would be interesting to compare offline and online training modes in this particular application. There might be an issue regarding the minimum and/or maximum number/s of training samples that can make online training useful, useless or destructive to the performance of a classifier. Other ways of combining GA, NN and FL can be explored further. These approaches provide tremendous promise to solving complex problems since they are all inspired by nature. It is also interesting to note that these three complement each other. One of the major issues in this study is the application of the GA-KSOM to select feature sets. The variation in the feature set combination as output of the GA-KSOM is indicative that the global optimum was not achieved. This might have been due to the fitness function as it dictates the error surface on which the search for solution/s is to take place. It is thus recommended to attempt to develop 'better' fitness functions. With regard to the features used, inclusion of other features which were not used in this study is certainly of significant interest. Previous studies have already dealt with colour and fractal features. In addition, morphological features should also be considered in future studies. Of particular interest is how to combine texture and morphological features together. Is it more effective to use morphological features prior to applying texture analysis or vice versa?

In conclusion, the aims of this research work have all been accomplished. Fuzzy logic, artificial neural networks, and genetic algorithms were effectively combined to implement feature selection and classification of colonic histopathological images. It was shown that the algorithms that were developed were effective enough in classifying colonic images as normal, dysplastic or adenomatous polyp, and cancerous cases. In evaluating the performances of the algorithms, a test dataset which was different from the training dataset was used. Comparison in terms of classification and misclassification between the algorithms developed here and the human experts who participated in the study showed almost similar results. This also demonstrated the robustness of the hybrid algorithms that were devised in this research.

REFERENCES

American Cancer Society (2008) *ACS :: How Is Colorectal Cancer Staged?* [online] available from <http://www.cancer.org/docroot/CRI/content/CRI_2_4_3X_How_is_colon_and_rectum_cancer_staged.asp> [14 November 2008]

Atlamazoglou V, Yova D, Kavantzas N, and Loukas S. (2001) 'Texture Analysis of Fluorescence Microscopic Images of Colonic Tissue Sections.' *Medical and Biological Engineering & Computing* 39, (2) 145-151.

Boland M V, Markey, M K, and Murphy R F (1998). 'Automated Recognition of Patterns Characteristics of Subcellular Structures in Fluorescence Microscopy Images'. *Cytometry* 33, (3) 366-375.

Boyle, P., and Langman, J., (2000) 'ABC of Colorectal Cancer: Epidemiology.' *BMJ*, 321, pp. 805-808

Cancer Research UK (2008a) *Cancer Research UK : CancerStats Key Facts on Bowel Cancer* [online] available from <http://info.cancerresearchuk.org/cancerstats/types/bowel/> [11 November 2008]

Cancer Research UK (2008b) *Cancer Research UK : CancerStats Key Facts on Bowel Cancer* [online] available from <http://info.cancerresearchuk.org/cancerstats/types/bowel/ > [11 November 2008]

Cancer Research UK (2008c) *Cancer Research UK : UK Bowel Cancer incidence statistics* [online] available from <http://info.cancerresearchuk.org/cancerstats/types/bowel/incidence/ > [11 November 2008]

Cordon, O, Herrera, F, Hoffmann, F, and Magdalena, L (2001). "Genetic Fuzzy Systems, Evolutionary Tuning and Learning of Fuzzy Knowledge Bases". World Scientific Publishing Co., Pte. Ltd.

Demir C and Yener B. (2005) *Automated cancer diagnosis based on histopathological images: a systematic survey, Technical report TR-05-09* [online]. Troy, USA: Department of Computer Science, Rensselaer Polytechnic Institute. Available from <http://www.cs.rpi.edu/research/pdf/05-09.pdf> [14 June 2008]

Dorundi, S., Bannarjea, A. (2006) 'Colorectal cancer: early diagnosis and predisposing causes.' *Surgery* 24, (4) 131-136.

Duda, R O, Hart, P E, and Stork D G (2001) 'Pattern Classification' 2/e. John Wiley & Sons, Inc.

Esgiar AN, Naguib RN G, Sharif BS, Bennett MK, and Murray A. (1999) 'Texture Descriptions and Classification for Pathological Analysis of Cancerous Colonic Mucosa.' *IEE Proc., Int. Conf. on Image Processing and its Applications*, '7th Interantional Conference on Image Processing and its Applications (IEE-IPA'99), (Conf. Publ. No. 465), vol. 1'. Held July 1999 at Manchester, UK: Institution of Electrical Engineers: 335-338.

Filippas, J, Amin, S A, Naguib, R N G, Bennett, M K. (2003a) 'A Parallel Implementation of a Genetic Algorithm for Colonic Tissue Image Classification'. Proc. Of the 4th Annual IEEE Conf. On Information Tehnology Applications in Biomedicine, UK, 2003.

Filippas, J, Arochena, H, Amin, S A, Naguib, R N G, Bennett, M K. (2003b) 'Comparison of Two AI Methods for Colonic Tissue Image Classification'. Proc. Of the 25th Annual International Conf. Of the IEEE EMBS. Cancun, Mexico. Sept. 17-21, 2003.

Fiscor, L, Varga, V S, Tagscherer, A, Tulassay, Z, Molnar, B (2008). 'Automated Classification of Inflammation in Colon Histological Sections Based on Digital

Microscopy and Advanced Image Analysis'. Cytometry Part A, 73A, pp 230-237, 2008.

Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2010a). 'Image classification of microscopic colonic images using textural properties and KSOM'. International Journal of Biomedical Engineering and Technology (IJBET), Vol. 3, Issue 3/4, pp 308 – 318, 2010.

Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2010b). 'Analysis of colonic histopathological images using pixel intensities and Hough Transform'. Philippine Science Letters, Vol. 3, No. 1, 2010.

Gonzalez, R C, Woods, R E, and Eddins, S L (2004). Digital Image Processing Using MATLAB. Pearson Prentice Hall.

Greene, F.L., Compton, C.C., Fritz, A.G., Shah, J.P., and Winchester, D.P. (eds.) (2006) *AJCC Cancer Staging Atlas*. Chicago, Il: Springer

Hamilton, P W, Allen, D C, Watt, P C H, Patterson, C C, Biggart, J D. (1987) 'Classification of Normal Colorectal Mucosa and Adenocarcinoma by Morphometry.' Histopathology, Vol. 11, Issue 9, pp901-911, Sept 1987.

Haralick R, Shanmugam K, and Dinstein I. (1973) 'Texture features for image classification.' IEEE Transactions on Systems, Man, and Cybernetics, SMC-3(6): 610-621.

Haralick RM and Shapiro LG, (1992) 'Computer and Robot Vision, Volume 2.' Addison-Wesley Publishing Company.

Holland, J H (1975). "Adaptation in Natural and Artificial Systems". Ann Arbor: University of Michigan Press.

Huang, S-J, Hung, C-C (1995). 'Genetic Algorithms Enhanced Kohonen's Neural Networks'. IEEE International Conference on Neural Networks (ICNN'95), Volume 2.

Jang, J-S (1993). “ANFIS: Adaptive-Network-Based Fuzzy Inference System”. IEEE Trans. on Systems, Man, and Cybernetics, vol. 23 pp. 665-685, May/June 1993.

Jang, J-S and Sun, C-T (1995). “Neuro-Fuzzy Modeling and Control”. Proc. of the IEEE, Vol. 83, No. 3, March 1995.

Karray FO and de Silva C, (2004) ‘Soft Computing and Intelligent Systems Design, Theory, Tools, and Applications.’ Pearson Education Limited.

Kiviluoto, K (1996). ‘Topology Preservation in Self-Organising Maps’. *Proceedings of the International Conference on Neural Networks (ICNN).* 294-299 (1996)

Kohonen T, (1990) ‘The Self-Organizing Map’. Proc. Of the IEEE, Vol. 78, No. 9: 1464-1480.

Kulkarni AD, (2001) ‘Computer Vision and Fuzzy-Neural Systems.’ Prentice Hall PTR

Masood, K, Rajpoot, N (2006). ‘Hyperspectral Colon Biopsy Classification into Normal and Malignant Categories’. Computer 2006.

Masood, K, Rajpoot, N (2008). ‘Spatial Analysis for Colon Biopsy Classification from Hyperspectral Imagery’. Annals of the BMVA, Vol. 2008, No. 4, pp 1-16, 2008.

Masood, K, Rajpoot, N (2009). ‘Texture Based Classification of Hyperspectral Colon Biopsy Samples Using CLBP’. Biomedical Imaging from Nano to Macro 2009 ISBI09 IEEE, Intl. Symposium on, pp 1011-1014, 2009.

Masood, K, Rajpoot, N, Rajpoot, K, Qureshi (2006). ‘Hyperspectral Colon Tissue Classification using Morphological Analysis,’. 2nd International Conf. on Emerging Technologies, IEEE-ICET 2006, Peshawar Pakistan, Nov 13-14, 2006.

McAndrew, A (2004). ‘Introduction to Digital Image Processing with MATLAB’. Thomson Learning, Inc.

Marghani, K, Dlay, S, Sharif, B, and Sims, A. (2003) 'Morphological and Texture features for cancers tissues microscopic images.' In Sonka, M and Fitzpatrick,

J. M. (eds.) *Medical Imaging 2003: Image Processing, Proceedings of SPIE Vol. 5032 (2003*), 'Medical Imaging 2003: Image Processing.' Held February 17, 2003 at San Diego, CA, USA: SPIE: 1757-1764.

National Cancer Institute (2008a) *NCI Visuals Online* [online] available from <http://visualsonline.cancer.gov/details.cfm?imageid=2512> [13 November 2008]

National Cancer Institute (2008b) *Understanding Cancer Series: Cancer - National Cancer Institute* [online] available from <http://www.cancer.gov/cancertopics/understandingcancer/cancer/Slide24> [3 December 2008]

National Cancer Institute (2008c) *NCI Visuals Online* [online] available from <http://visualsonline.cancer.gov/details.cfm?imageid=2351> [13 November 2008]

National Cancer Institute (2008d) *Understanding Cancer Series: Cancer - National Cancer Institute* [online] available from <http://www.cancer.gov/cancertopics/understandingcancer/cancer/Slide2> [3 December 2008]

National Cancer Institute (2008e) *Understanding Cancer Series: Cancer - National Cancer Institute* [online] available from <http://www.cancer.gov/cancertopics/understandingcancer/cancer/Slide4> [3 December 2008]

National Cancer Institute (2008f) *NCI Visuals Online: Image Details* [online] available from <http://visuals.nci.nih.gov/details.cfm?imageid=1781> [12 November 2008]

National Cancer Institute (2008g) *NCI Dictionary of Cancer Terms* [online] available from

<http://www.cancer.gov/Common/PopUps/popDefinition.aspx?term=differentiation&version=Patient&language=English: > [12 November2008]

National Cancer Institute (2008h) *NCI Visuals Online: Image Details* [online] available from <http://visualsonline.cancer.gov/details.cfm?imageid=7180> [Nov 12, 2008]

National Cancer Institute (2008i) *NCI Visuals Online: Image Details* [online] available from <http://visualsonline.cancer.gov/details.cfm?imageid=7181> [12 November 2008]

Ngelangel, C. A., Wang, E. H. M. (2002) 'Cancer and the Philippine Cancer Control Program.' *Japanese Journal of Oncology*, Vol. 32, Supp. 1, pp. S52-S61

NHS (2008) *NHS Bowel Cancer Screening Programme* [online] available from <http://www.cancerscreening.nhs.uk/bowel/index.html> [12 November 2008]

Nwoye E, Woo WL, and Dlay SS (2004) 'Fuzzy-Neural Machine with Image Feature Extraction for Colorectal Cancer Diagnosis.' In Villanieva, J. (ed.) *Proc. of the Fourth IASTED International Conference, Visualization, Imaging, and Image Processing,* 'Fourth IASTED International Conference, Visualization, Imaging, and Image Processing (VIIP'04).' Held September 6-8, 2004 at Marbella, Spain: ACTA Press: 304-308.

Patel, A C, and Markey, M K (2005). 'Comparison of Three-Class Classification Performance Metrics: A Case Study in Breast Cancer CAD'. Medical Imaging 2005: Image Perception, Observer Performance, and Technology Assessment, Proceedings of SPIE Vol. 5749, Eckstein and Jiang (eds.), SPIE, Bellingham, WA 2005.

Polani, D., Uthman, T. (1992). 'Adaptation of Kohonen Feature Map Topologies by Genetic Algorithms'. Parallel Problem Solving from Nature, 2, page 421-429, Manner, A. And Manderick. Elsevier Science Publishers B.V. (editors), September 28-30, 1992.

Rajpoot, K, Rajpoot, N. (2004) 'SVM Optimization for Hyperspectral Tissue Cell Classification'. Processing, Vol. 3217/2004, pp 829-837.

Rajpoot K M, Rajpoot, N M, Turner, M J. (2006). 'Hyperspectral Colon Tissue Cell Classification'. Proc. of International Conference on Emerging Technologies 2006.

Shuttleworth JK, Todman A G, Naguib R N G, Newman B M, and Bennett M K. (2002a) 'Colour Texture Analysis using Co-occurrence Matrices for Classification of Colon Images.' *Proceedings of the 2002 IEEE Canadian Conference on Electrical & Computer Engineering*, 'IEEE Canadian Conference on Electrical & Computer Engineering.' Held May 12-15, 2002 at Winnipeg, Canada: IEEE: 1134-1139.

Shuttleworth JK, Todman AG, Naguib RNG, Newman RM and Bennett MK. (2002b) 'Multiresolution Colour Texture Analysis for Classifying Colon Cancer Images.' *Proc. Int. Conf. IEEE Eng. in Med. and Biol. Soc.*, '24th Annual International Conference of the IEEE Engineering in Medicine and Biology Society.' Held October 23-26, 2002 at Houston, USA: IEEE EMBS:1118-1119.

Shuttleworth JK, Todman AG, Naguib RNG, Newman RM and Bennett MK (2005) 'Enhancing Feature Extraction from Colon Microscopy Images Using Colourspace Rotation.' In Mirmehdi, M. (ed.) *Proc. of the 9th Medical Image Understanding and Analysis Conf.*, 'Medical Image Understanding and Analysis 2005.' Held July 19-20 2005 at University of Bristol. Bristol, UK: BMVA Press: 11-14.

Tjoa M P and Krishnan S M (2002) 'Texture-based Quantitative Characterization and Analysis of Colonoscopic Images,' *Proceedings of the 2nd Joint EMBS/BMES Conference, Vol. 2*, 'Second Joint EMBS/BMES Conference.' Held October 23-26, 2002 at Houston, USA: EMBS/BMES: 1090-1091

Uriarte, E A and Martin, F D (2005). 'Topology Preservation in SOM'. *International Journal of Mathematical and Computer Sciences* 1:1 2005

World Health Organization (2009) *WHO Cancer* [online] available from <http://www.who.int/mediacentre/factsheets/fs297/en/index.html> [14 February 2009]

Ye, Nong (ed.) (2003). *Handbook of Data Mining*. Lawrence Erlbaum Associates, Inc.

Zucker, S W, and Terzopoulus, D (1980). 'Finding Structure in Co-Occurrence Matrices for Texture Analysis.' *Computer Graphics and Image Processing 12, pp. 286-308.*

APPENDICES

A.1 TRAINING AND TEST *.data:

A *.data file presents data in a row-by-row format. Each row represents one data point while each column represents the value of the corresponding property as defined in the header portion of the file. The last column of the data lists the class or category of each data point in each row where C means cancerous, N means normal, and P means adenomatous polyp. In this research, the header portion contains four lines wherein the last line lists the names of the properties. Below is a print-out of the *.data training data file for the 1200x1600 pixels image size with 16 quantisation levels.

"trainingData1200x1600.data" – training data for image size of 1200x1600 pixels, 16 quantisation levels:

```
15
#l image classification
## N - normal, P - adenomatous polyp, C - cancerous
#n ASM contrast mean variance correlation IDM sumAverage sumEntropy sumVariance entropy differenceVariance differenceEntropy IMC12 IMC13 MCC
0.125007 0.170120 3.855573 3.060108 0.972204 0.918264 7.711145 0.978309 57.401400 1.034187 0.143034 0.198247 -0.704622 0.822079 0.970251 C
0.124931 0.199096 3.405919 2.705756 0.963209 0.904642 6.811838 0.983556 44.592807 1.049129 0.162189 0.217762 -0.665464 0.805594 0.958287 C
0.138221 0.183533 4.345571 5.137895 0.982139 0.912161 8.691141 0.965054 80.060472 1.025060 0.152208 0.207518 -0.702409 0.818939 0.986888 C
0.132547 0.218881 3.735606 4.187050 0.973862 0.896070 7.471212 0.991344 58.518017 1.063340 0.174908 0.229827 -0.655112 0.803390 0.976806 C
0.122801 0.194368 4.300907 2.909892 0.966602 0.906227 8.601814 0.990843 69.372072 1.054187 0.158767 0.214852 -0.673203 0.810773 0.963386 C
0.131815 0.139745 4.014158 2.853554 0.975514 0.931369 8.028316 0.956872 61.279785 1.001157 0.120790 0.175805 -0.737615 0.830537 0.976308 C
0.131505 0.179230 3.443315 2.487551 0.963975 0.912609 6.886630 0.968746 44.792329 1.026201 0.148421 0.204899 -0.681125 0.808507 0.956967 C
0.126584 0.200688 4.584548 2.278088 0.955952 0.902898 9.169097 0.965892 76.204245 1.031240 0.162552 0.218892 -0.652984 0.795558 0.955236 C
0.136288 0.176692 4.543828 2.169902 0.959286 0.914578 9.087655 0.942316 74.849463 1.000070 0.147171 0.203045 -0.678149 0.801230 0.960782 C
0.113054 0.192536 4.826225 2.844043 0.966151 0.905646 9.652450 1.012822 85.826806 1.074035 0.156684 0.213693 -0.677461 0.817303 0.964075 C
```

0.117773 0.197195 4.339520 2.459032 0.959904 0.905708 8.679040 0.990840 68.747353 1.056239 0.161087 0.216507 -0.667520 0.808372 0.954541 C
0.136979 0.188699 4.086967 2.731509 0.965459 0.908863 8.173934 0.945746 62.984031 1.007235 0.155084 0.211162 -0.668952 0.798198 0.971315 C
0.135454 0.156499 4.516894 2.515723 0.968896 0.923324 9.033789 0.948053 75.285515 0.997929 0.132821 0.188732 -0.708672 0.816272 0.969519 C
0.160475 0.158068 3.845441 2.183135 0.963798 0.922374 7.690883 0.905987 54.609289 0.956062 0.133819 0.189925 -0.689046 0.796749 0.965537 C
0.115947 0.230888 4.396012 2.404681 0.951992 0.890075 8.792024 1.001128 70.085902 1.077955 0.181742 0.236812 -0.629761 0.793277 0.947732 C
0.118736 0.233967 4.280716 2.296801 0.949067 0.887090 8.561431 1.004113 66.066293 1.080471 0.182357 0.238465 -0.620419 0.788739 0.936887 C
0.127576 0.209723 4.094944 2.136464 0.950918 0.898245 8.189888 0.962228 60.575205 1.030166 0.167884 0.224456 -0.637662 0.786934 0.953110 C
0.098952 0.221130 4.719933 3.981506 0.972230 0.894513 9.439866 1.056137 85.991792 1.129051 0.175903 0.231143 -0.669497 0.824240 0.978181 C
0.123224 0.184730 4.255114 2.527596 0.963457 0.911195 8.510228 0.970122 66.778843 1.030963 0.152762 0.208428 -0.679115 0.808629 0.963835 C
0.109173 0.205452 4.996712 2.851197 0.963971 0.899531 9.993424 1.023374 91.661146 1.088922 0.164773 0.221804 -0.665134 0.814162 0.961687 C
0.123805 0.212671 5.067305 2.566430 0.958567 0.897279 10.134610 0.980806 93.845192 1.049980 0.169968 0.226247 -0.645387 0.795637 0.950482 C
0.161354 0.149878 4.433950 2.061266 0.963644 0.926724 8.867900 0.914846 71.346252 0.962774 0.128237 0.183690 -0.707703 0.807365 0.954796 C
0.095909 0.203570 4.794450 3.805530 0.973253 0.903125 9.588900 1.067210 87.637743 1.134864 0.165394 0.220472 -0.693473 0.836866 0.969356 C
0.105385 0.216202 4.761609 3.128214 0.965443 0.896887 9.523219 1.042801 84.214145 1.114219 0.172984 0.228273 -0.664901 0.818895 0.956322 C
0.117281 0.182762 4.670967 2.828605 0.967694 0.911274 9.341934 1.001563 80.693432 1.060676 0.150958 0.207266 -0.691106 0.821054 0.959460 C
0.103928 0.246038 5.068270 2.792731 0.955950 0.882871 10.136539 1.043306 93.611781 1.124847 0.190237 0.245120 -0.626526 0.801205 0.943095 C
0.121699 0.191726 4.820646 2.857705 0.966455 0.908811 9.641293 0.995913 85.981685 1.059617 0.157894 0.212829 -0.682206 0.816460 0.957053 C
0.111319 0.227970 4.983716 2.629065 0.956644 0.889712 9.967432 1.020286 90.339712 1.094281 0.178772 0.235112 -0.635752 0.799840 0.944672 C
0.158780 0.143511 5.007224 3.026071 0.976288 0.929313 10.014449 0.918408 94.698737 0.963539 0.123424 0.178858 -0.732768 0.820048 0.976790 C
0.099895 0.177622 4.913013 4.725445 0.981206 0.913778 9.826026 1.058041 95.601721 1.115247 0.147587 0.203739 -0.721323 0.846151 0.981501 C
0.115671 0.172971 4.180932 4.081088 0.978808 0.915758 8.361864 1.016864 70.100412 1.072317 0.144331 0.200556 -0.716797 0.835936 0.978435 C
0.114485 0.168530 4.605965 4.062595 0.979258 0.916529 9.211931 1.028022 83.058219 1.080188 0.140572 0.197434 -0.722863 0.840089 0.980379 C
0.124957 0.148207 5.097872 2.963817 0.974997 0.926669 10.195744 0.986867 96.510471 1.032977 0.126621 0.182524 -0.735773 0.836679 0.977786 C
0.134598 0.129906 4.754457 3.030181 0.978565 0.935702 9.508914 0.948990 85.263118 0.989388 0.113312 0.167880 -0.753783 0.835519 0.976537 C
0.096985 0.182116 4.840759 4.360399 0.979117 0.911840 9.681518 1.065858 91.489066 1.124881 0.150686 0.206787 -0.715729 0.845892 0.976075 C
0.083527 0.265067 5.041813 4.493311 0.970504 0.878671 10.083627 1.110879 98.218392 1.201735 0.204359 0.254603 -0.647223 0.826922 0.969619 C
0.083341 0.252221 5.760697 4.823874 0.973857 0.883844 11.521395 1.110973 127.420153 1.196999 0.196701 0.248006 -0.663022 0.834022 0.972020 C
0.079522 0.271742 5.822017 4.640214 0.970719 0.876370 11.644034 1.122340 128.995163 1.215798 0.208572 0.258010 -0.649683 0.830750 0.966005 C
0.087383 0.296656 5.845358 5.326976 0.972155 0.868629 11.690715 1.114026 132.877608 1.216915 0.224622 0.269675 -0.637267 0.824875 0.968993 C
0.081850 0.284038 5.575007 5.328936 0.973349 0.873065 11.150015 1.118079 121.671433 1.216194 0.217012 0.263710 -0.645993 0.829001 0.971755 C
0.073822 0.309526 5.678827 4.685198 0.966968 0.863101 11.357653 1.136110 122.911217 1.243718 0.231267 0.275548 -0.624117 0.822792 0.961611 C
0.078433 0.309260 5.020430 4.561055 0.966098 0.862326 10.040860 1.123016 97.462903 1.230102 0.230302 0.275521 -0.616910 0.816593 0.962850 C
0.115897 0.196006 5.415930 3.231711 0.969675 0.905728 10.831860 0.998411 109.427546 1.062609 0.159987 0.215850 -0.677842 0.815179 0.972876 C
0.178718 0.149033 4.948939 3.222346 0.976875 0.928003 9.897878 0.879767 94.066672 0.928325 0.128057 0.182739 -0.721868 0.806298 0.980126 C
0.150299 0.165048 4.922131 2.970581 0.972220 0.920101 9.844263 0.925605 91.259728 0.979256 0.139232 0.194825 -0.705539 0.810290 0.972946 C
0.156784 0.171263 4.875506 2.746276 0.968819 0.916408 9.751013 0.902702 89.106440 0.957507 0.143084 0.199419 -0.686836 0.796092 0.972927 C
0.136643 0.199941 4.931207 2.589006 0.961387 0.903920 9.862413 0.947102 89.638853 1.012770 0.162515 0.218388 -0.657247 0.793647 0.960396 C
0.119492 0.213509 5.089341 3.033317 0.964806 0.896997 10.178683 1.002768 96.117175 1.072282 0.170554 0.226750 -0.655839 0.805940 0.969921 C
0.165561 0.190002 5.226434 2.796782 0.966032 0.908669 10.452868 0.922281 101.829209 0.984559 0.156188 0.211964 -0.669729 0.793583 0.968616 C
0.136296 0.183289 4.857984 3.335881 0.972528 0.912960 9.715969 0.965789 89.725889 1.026903 0.152448 0.207127 -0.693987 0.815303 0.968269 C
0.113823 0.176548 4.270378 2.903557 0.969598 0.914663 8.540756 0.997947 68.331647 1.055544 0.147083 0.202926 -0.699515 0.823929 0.967275 C

0.116441 0.191023 5.026137 2.898805 0.967051 0.907208 10.052275 1.010454 93.158714 1.072155 0.156244 0.212747 -0.680871 0.818436 0.961494 C
0.187538 0.145695 5.181348 3.224797 0.977410 0.930128 10.362695 0.887190 102.538682 0.935143 0.125888 0.179816 -0.728172 0.811154 0.974458 C
0.126722 0.207273 4.451558 1.991849 0.947970 0.899596 8.903117 0.963592 70.796181 1.031017 0.166515 0.222977 -0.636167 0.786247 0.937610 C
0.100710 0.223237 4.870316 2.974100 0.962470 0.892743 9.740633 1.047972 87.235519 1.121250 0.176595 0.232441 -0.654214 0.814957 0.956960 C
0.113265 0.208776 4.616690 2.661026 0.960771 0.899922 9.233380 0.991464 78.364500 1.060320 0.168137 0.223853 -0.657300 0.803942 0.954802 C
0.148402 0.130480 3.964063 3.214754 0.979706 0.935478 7.928126 0.914651 61.917369 0.955246 0.113765 0.168345 -0.745011 0.823734 0.983591 C
0.134376 0.152238 4.064783 2.825583 0.973061 0.924884 8.129567 0.948791 62.713634 0.996424 0.129567 0.185596 -0.715908 0.819476 0.970725 C
0.132515 0.128908 4.009126 3.194419 0.979823 0.936177 8.018253 0.950265 62.605218 0.990314 0.112561 0.167049 -0.754393 0.835948 0.981101 C
0.145265 0.131234 4.279535 3.386267 0.980623 0.935597 8.559070 0.921800 71.741732 0.963379 0.114538 0.168851 -0.749987 0.827956 0.984279 C
0.148229 0.128336 4.326615 3.957227 0.983785 0.936685 8.653230 0.937199 75.237716 0.977334 0.112229 0.166529 -0.757546 0.834701 0.982645 C
0.112840 0.176694 4.549868 4.330751 0.979600 0.913707 9.099736 1.025556 82.338693 1.081802 0.146671 0.203176 -0.713053 0.835845 0.981975 C
0.135518 0.144438 3.908530 2.346620 0.969224 0.928769 7.817060 0.943465 56.488353 0.988809 0.124048 0.179583 -0.721221 0.820125 0.963006 C
0.118403 0.206880 4.385180 2.711858 0.961856 0.901054 8.770360 0.983163 71.280991 1.051613 0.167123 0.222657 -0.659147 0.802964 0.962578 C
0.109405 0.212727 4.382047 2.690177 0.960462 0.899264 8.764095 1.006491 70.728406 1.077790 0.171376 0.226027 -0.656709 0.807298 0.955138 C
0.095591 0.249059 4.478567 3.199448 0.961078 0.881994 8.957135 1.063394 74.859884 1.146250 0.192327 0.246764 -0.635357 0.810121 0.954420 C
0.134795 0.197153 4.488728 3.605645 0.972661 0.907431 8.977456 0.963575 78.447708 1.029830 0.162131 0.216003 -0.684818 0.811221 0.975310 C
0.114826 0.207981 4.303021 2.854655 0.963572 0.900845 8.606042 0.993783 69.157131 1.062828 0.168013 0.223293 -0.661333 0.806530 0.962252 C
0.115469 0.213275 4.072932 3.031250 0.964821 0.897522 8.145864 0.993345 63.070245 1.063094 0.170698 0.226626 -0.657756 0.804622 0.965259 C
0.121331 0.196854 4.253161 2.882884 0.965858 0.904421 8.506322 0.977844 68.012670 1.041376 0.159949 0.216510 -0.670779 0.806624 0.967313 C
0.155773 0.163359 7.180863 3.448607 0.976315 0.919842 14.361726 0.944395 193.655846 0.995849 0.137495 0.193808 -0.706370 0.814645 0.960864 N
0.237670 0.150950 7.638685 3.280879 0.976995 0.928255 15.277370 0.856471 220.934890 0.906176 0.130003 0.183570 -0.707552 0.793969 0.958546 N
0.164912 0.158337 7.167472 2.697152 0.970647 0.923647 14.334944 0.920037 190.590005 0.971350 0.134730 0.189756 -0.701250 0.806920 0.953750 N
0.145767 0.172355 6.965958 3.226280 0.973289 0.914873 13.931917 0.966426 180.836703 1.020030 0.143249 0.200163 -0.693985 0.814030 0.961275 N
0.145558 0.195305 7.125833 3.680418 0.973467 0.906751 14.251666 0.969887 190.932037 1.033735 0.159974 0.215212 -0.675909 0.807994 0.955289 N
0.163343 0.199016 7.269472 3.612272 0.972453 0.907532 14.538944 0.943862 199.076330 1.010547 0.163942 0.216698 -0.671845 0.800743 0.952003 N
0.178016 0.160868 7.380035 3.144612 0.974422 0.922039 14.760070 0.906709 204.333190 0.958525 0.136299 0.191780 -0.700133 0.803339 0.961256 N
0.151816 0.162140 7.059696 3.887406 0.979145 0.923415 14.119393 0.958958 188.584537 1.012918 0.138219 0.191746 -0.723223 0.826365 0.964336 N
0.210204 0.159330 7.220108 4.443964 0.982073 0.925787 14.440216 0.901108 200.923952 0.954706 0.136757 0.188998 -0.728189 0.815609 0.968621 N
0.161191 0.196210 7.038120 4.428676 0.977848 0.911329 14.076239 0.963922 189.451367 1.031099 0.163636 0.213457 -0.695783 0.817082 0.965725 N
0.223727 0.140103 7.545030 3.076521 0.977230 0.933674 15.090060 0.861303 214.623523 0.907812 0.122175 0.174745 -0.729374 0.805343 0.965232 N
0.149482 0.172766 7.155566 3.237841 0.973321 0.918233 14.311132 0.949649 191.307814 1.007186 0.145517 0.199599 -0.701352 0.814673 0.959213 N
0.141765 0.196415 6.896658 4.496056 0.978157 0.909834 13.793315 0.993169 181.631565 1.059861 0.162922 0.214408 -0.697481 0.824138 0.967801 N
0.238735 0.136684 7.634447 3.205917 0.978683 0.935440 15.268894 0.856351 220.408387 0.901779 0.119685 0.171815 -0.732164 0.804998 0.961009 N
0.190617 0.252552 7.309306 4.743478 0.973379 0.890203 14.618613 0.973810 204.902005 1.060633 0.201893 0.245736 -0.642878 0.796843 0.960643 N
0.129575 0.229925 7.692237 5.651149 0.979657 0.898259 15.384474 1.036719 228.232751 1.114606 0.186712 0.233403 -0.679830 0.826864 0.961396 N
0.150477 0.226098 7.036046 4.973524 0.977270 0.900663 14.072092 1.011955 190.235176 1.089789 0.184794 0.230709 -0.676481 0.820225 0.964808 N
0.157415 0.218292 6.955101 5.109486 0.978639 0.904331 13.910202 0.990112 187.148390 1.065566 0.179946 0.225768 -0.684403 0.819175 0.965411 N
0.159986 0.210625 7.040631 5.001335 0.978943 0.906711 14.081262 0.992387 191.113361 1.064819 0.174306 0.221533 -0.689083 0.821294 0.965793 N
0.145796 0.232208 7.004155 5.095003 0.977212 0.898272 14.008310 1.021271 188.810974 1.101168 0.188846 0.234207 -0.672821 0.820688 0.965148 N
0.164053 0.219365 6.992374 5.153291 0.978716 0.904082 13.984747 0.985477 189.374822 1.061355 0.180787 0.226325 -0.682926 0.817545 0.965368 N
0.085372 0.269194 6.330756 4.899173 0.972527 0.877191 12.661512 1.130599 152.289467 1.222807 0.206922 0.256694 -0.650487 0.832601 0.964514 N

0.106754 0.215971 6.475241 4.382275 0.975359 0.900118 12.950482 1.058901 158.722830 1.132503 0.174979 0.227254 -0.687475 0.833927 0.967027 N
0.120183 0.222095 6.932370 4.107830 0.972967 0.895171 13.864740 1.051354 180.392090 1.125365 0.177266 0.231539 -0.670286 0.824173 0.960846 N
0.097485 0.225098 6.089480 5.179206 0.978269 0.897020 12.178960 1.085846 143.548901 1.163015 0.181356 0.232286 -0.689241 0.840475 0.973274 N
0.118137 0.190540 6.500400 4.763401 0.980000 0.909806 13.000801 1.039721 161.930483 1.103053 0.157388 0.211731 -0.708733 0.838420 0.973360 N
0.101536 0.235695 6.157711 4.034025 0.970787 0.891168 12.315422 1.078760 142.162960 1.158927 0.187000 0.238877 -0.663730 0.827201 0.959629 N
0.162925 0.184710 6.943569 4.635254 0.980076 0.915925 13.887138 0.971618 185.166957 1.035035 0.155500 0.206049 -0.712589 0.826397 0.969166 N
0.157992 0.210004 7.240336 3.431605 0.969401 0.902004 14.480672 0.962126 196.267499 1.032679 0.170672 0.223890 -0.663093 0.801350 0.953938 N
0.087818 0.260329 5.806207 3.625001 0.964093 0.879884 11.612415 1.094259 124.871276 1.182867 0.200999 0.252254 -0.638688 0.819285 0.956630 N
0.092166 0.254778 6.341515 3.603913 0.964653 0.881645 12.683030 1.083490 148.710212 1.169730 0.197314 0.249447 -0.640478 0.817912 0.954814 N
0.104926 0.230953 6.569354 3.871773 0.970175 0.893660 13.138709 1.056467 161.236700 1.134928 0.184416 0.236046 -0.666004 0.823920 0.958812 N
0.094425 0.270455 6.487809 4.017915 0.966344 0.877430 12.975618 1.085485 157.176486 1.178302 0.208277 0.257128 -0.633719 0.816103 0.955574 N
0.099484 0.236597 6.465413 3.600156 0.967141 0.890298 12.930825 1.061851 155.036585 1.142143 0.187195 0.239469 -0.657132 0.821077 0.959008 N
0.109867 0.252201 6.793315 3.885239 0.967544 0.884165 13.586631 1.052958 172.381703 1.138744 0.196935 0.247880 -0.642705 0.813081 0.955019 N
0.112866 0.514436 9.698976 7.658035 0.966412 0.825325 19.397952 1.139485 363.489330 1.305643 0.372798 0.340465 -0.554827 0.796293 0.951646 N
0.064579 0.553925 9.699964 7.757584 0.964298 0.802320 19.399927 1.215994 361.131826 1.398755 0.376327 0.356439 -0.545029 0.806528 0.953251 N
0.066889 0.526630 9.627942 7.681400 0.965720 0.812786 19.255884 1.210762 355.825383 1.384536 0.367114 0.347472 -0.555116 0.809867 0.954291 N
0.088843 0.492100 9.331408 7.618488 0.967704 0.817092 18.662815 1.175002 335.805475 1.339685 0.342564 0.339646 -0.558475 0.804216 0.958346 N
0.063425 0.622198 9.620634 7.246489 0.957069 0.790328 19.241268 1.221011 353.093425 1.422181 0.417295 0.373343 -0.518173 0.794462 0.939965 N
0.065920 0.511571 9.408532 7.021513 0.963571 0.812209 18.817064 1.216438 337.356499 1.388139 0.353225 0.344404 -0.553806 0.809953 0.952288 N
0.067566 0.411382 9.655391 5.825843 0.964693 0.839880 19.310782 1.198789 350.936279 1.340799 0.299019 0.312024 -0.593637 0.823885 0.949870 N
0.037838 0.533860 8.523579 12.872825 0.979264 0.801602 17.047158 1.367180 296.819158 1.545983 0.358029 0.351685 -0.600244 0.857298 0.966844 N
0.036886 0.581139 8.835526 13.285606 0.978129 0.791254 17.671053 1.375081 318.119987 1.565606 0.383736 0.363851 -0.587067 0.853351 0.965894 N
0.038182 0.591546 8.913883 14.376282 0.979426 0.793651 17.827766 1.376328 327.563386 1.567485 0.396309 0.365403 -0.592061 0.856039 0.968276 N
0.043798 0.515447 8.631708 14.513783 0.982243 0.812216 17.263416 1.346722 310.880851 1.516287 0.356669 0.345154 -0.614123 0.860041 0.972595 N
0.038041 0.839119 8.614987 17.142190 0.975525 0.760948 17.229974 1.373233 319.165875 1.609025 0.551679 0.416015 -0.549940 0.839818 0.967157 N
0.040409 0.634899 9.350212 14.391124 0.977941 0.785050 18.700424 1.363827 357.487194 1.564862 0.420167 0.376537 -0.577148 0.848368 0.968097 N
0.040359 0.533015 8.465533 13.411395 0.980128 0.803977 16.931067 1.361503 295.523893 1.539195 0.360570 0.351614 -0.604993 0.858728 0.967757 N
0.038500 0.610034 8.726281 14.928534 0.979568 0.791680 17.452563 1.376226 317.552709 1.570508 0.409244 0.369426 -0.589176 0.855005 0.969367 N
0.076949 0.443387 9.749791 6.308254 0.964857 0.831228 19.499583 1.182772 360.295194 1.334459 0.317368 0.323404 -0.574760 0.812584 0.949862 N
0.082379 0.422724 9.614056 6.179523 0.965796 0.835897 19.228112 1.174241 350.237632 1.319739 0.304532 0.316716 -0.580798 0.813540 0.951234 N
0.060597 0.522820 9.779406 6.921290 0.962231 0.806827 19.558812 1.221020 363.436981 1.397948 0.355633 0.348891 -0.549936 0.809121 0.948200 N
0.081670 0.449903 9.489260 6.080986 0.963007 0.826433 18.978521 1.169544 341.033687 1.323889 0.317307 0.326786 -0.564503 0.805083 0.949145 N
0.075120 0.578891 9.676215 8.030394 0.963956 0.801390 19.352431 1.211560 360.633865 1.399091 0.396592 0.361843 -0.538459 0.802928 0.945493 N
0.065421 0.583103 9.768058 8.296798 0.964860 0.798005 19.536116 1.226919 367.830779 1.416937 0.395264 0.363927 -0.542185 0.807944 0.947362 N
0.097625 0.436893 9.607220 6.265907 0.965137 0.831506 19.214439 1.140514 351.293519 1.289945 0.311915 0.322196 -0.567048 0.800454 0.947826 N
0.089687 0.471914 9.743796 6.615297 0.964332 0.820513 19.487591 1.163341 361.767425 1.322652 0.329293 0.333780 -0.554834 0.799393 0.950505 N
0.126005 0.420027 9.853775 7.147407 0.970617 0.832769 19.707549 1.118418 373.725419 1.260639 0.298461 0.317782 -0.571658 0.797975 0.950447 N
0.106844 0.429321 9.605948 7.261018 0.970437 0.828895 19.211895 1.149738 354.856284 1.295878 0.302111 0.321067 -0.574056 0.805637 0.953017 N
0.090486 0.477733 9.543690 6.794435 0.964844 0.814757 19.087381 1.170227 347.724419 1.332389 0.326978 0.336669 -0.551632 0.799317 0.945686 N
0.089516 0.552285 9.853778 6.769289 0.959207 0.806208 19.707556 1.173693 370.028927 1.354270 0.380488 0.354572 -0.529618 0.790240 0.935548 N
0.095974 0.536220 9.692238 6.639484 0.959619 0.811271 19.384476 1.157216 358.254704 1.333521 0.373478 0.350047 -0.533402 0.788861 0.935751 N

0.146660 0.529899 10.175122 5.188438 0.948935 0.819640 20.350244 1.061256 392.288919 1.234429 0.378922 0.347420 -0.524900 0.766092 0.927371 N
0.054277 0.611029 9.815049 5.496508 0.944417 0.776328 19.630098 1.220618 360.283950 1.425362 0.386004 0.372499 -0.496656 0.782361 0.922542 N
0.047015 0.775402 9.638678 7.368184 0.947382 0.746915 19.277355 1.265825 353.112569 1.510432 0.472456 0.408590 -0.475472 0.782069 0.928263 N
0.048193 0.704767 9.453243 8.710853 0.959547 0.770891 18.906485 1.295191 344.296322 1.519863 0.456447 0.391823 -0.519066 0.810138 0.940799 N
0.043228 0.692974 9.150424 8.954343 0.961305 0.769617 18.300847 1.311256 323.770634 1.534736 0.444360 0.389657 -0.522989 0.814696 0.946162 N
0.038034 0.730092 9.255897 10.163313 0.964082 0.763774 18.511793 1.333577 335.014273 1.564149 0.465557 0.397324 -0.527188 0.821424 0.951577 N
0.035794 0.755427 9.232273 9.908332 0.961879 0.757834 18.464546 1.340894 332.097341 1.577894 0.476280 0.402309 -0.517418 0.817611 0.946964 N
0.121889 0.165021 4.523701 2.509482 0.967120 0.918288 9.047403 0.968432 75.142667 1.019673 0.138227 0.194970 -0.698232 0.816231 0.961190 P
0.124776 0.155236 4.444859 2.460118 0.968450 0.923586 8.889718 0.958963 72.582103 1.007904 0.131756 0.187849 -0.711785 0.819797 0.963366 P
0.111503 0.192586 4.626555 2.741655 0.964878 0.906309 9.253110 1.010998 78.706439 1.073053 0.157148 0.213758 -0.677083 0.816792 0.957171 P
0.125627 0.161037 4.495616 2.564199 0.968599 0.920438 8.991233 0.956327 74.655478 1.006627 0.135615 0.192128 -0.704246 0.815899 0.965551 P
0.120169 0.166121 4.601143 2.792539 0.970256 0.918917 9.202287 0.973013 78.724979 1.026267 0.139609 0.195745 -0.705317 0.820634 0.965325 P
0.129961 0.163562 4.992428 2.977017 0.972529 0.920014 9.984855 0.975225 92.917948 1.027422 0.137780 0.193924 -0.708577 0.822648 0.966877 P
0.123235 0.151863 4.580502 2.831851 0.973187 0.924642 9.161004 0.964242 78.362442 1.011123 0.129090 0.185290 -0.721123 0.825167 0.968556 P
0.096855 0.250756 4.950739 6.978780 0.982034 0.887403 9.901478 1.108660 104.978000 1.193928 0.198112 0.246092 -0.675700 0.839403 0.979118 P
0.065255 0.314380 5.465064 7.305941 0.978485 0.867234 10.930129 1.202908 123.528217 1.312290 0.239505 0.275702 -0.654589 0.849272 0.973860 P
0.075006 0.301557 5.513019 6.296554 0.976054 0.867965 11.026038 1.167572 122.074005 1.271811 0.228492 0.271398 -0.646714 0.838902 0.968805 P
0.126433 0.200468 5.009201 7.400086 0.986455 0.908004 10.018403 1.027464 110.236862 1.095022 0.165597 0.217151 -0.712086 0.838258 0.984258 P
0.125996 0.182176 5.166548 7.307007 0.987534 0.918350 10.333095 1.024009 115.704942 1.086732 0.154475 0.203190 -0.733900 0.846589 0.986928 P
0.090795 0.269190 5.040839 7.956493 0.983084 0.881585 10.081678 1.132181 111.650276 1.223440 0.210525 0.255098 -0.672044 0.842832 0.982276 P
0.077136 0.223331 5.486965 5.855394 0.980929 0.897106 10.973929 1.157590 119.558763 1.233587 0.179774 0.231325 -0.701335 0.858170 0.973352 P
0.089527 0.225153 5.668601 4.128550 0.972732 0.893796 11.337202 1.089859 121.297085 1.165311 0.179130 0.233311 -0.672882 0.833130 0.966817 P
0.085174 0.245482 5.905458 4.646218 0.973583 0.887103 11.810915 1.100812 133.045715 1.184687 0.193007 0.244276 -0.664879 0.832685 0.969237 P
0.080001 0.230179 6.258450 4.945905 0.976730 0.890333 12.516899 1.132226 149.164221 1.208213 0.181276 0.236343 -0.681365 0.844617 0.972108 P
0.094427 0.195224 5.968681 4.025356 0.975751 0.905016 11.937363 1.066687 134.077786 1.129420 0.158803 0.215435 -0.698425 0.838413 0.970609 P
0.090116 0.228260 6.138478 4.290659 0.973400 0.892365 12.276957 1.084692 142.201179 1.161002 0.180983 0.235119 -0.673930 0.832587 0.967033 P
0.101023 0.199841 6.431056 4.259817 0.976544 0.903733 12.862111 1.072134 155.842980 1.137279 0.162301 0.218276 -0.695960 0.838701 0.970207 P
0.099738 0.186790 5.496952 3.714247 0.974855 0.909942 10.993905 1.053528 113.481285 1.114563 0.153946 0.209793 -0.703418 0.837793 0.971148 P
0.080474 0.264288 5.657162 5.407458 0.975563 0.879299 11.314325 1.145047 124.779760 1.235189 0.204161 0.253958 -0.661380 0.839947 0.973533 P
0.087209 0.268502 5.650779 5.518477 0.975672 0.881716 11.301558 1.113918 125.593405 1.207642 0.209996 0.255079 -0.661955 0.835434 0.971754 P
0.118266 0.243122 6.621042 4.769514 0.974513 0.893053 13.242085 1.048090 167.528453 1.132934 0.195267 0.240991 -0.667696 0.824423 0.964945 P
0.066198 0.329733 5.842814 5.102145 0.967687 0.855124 11.685628 1.183035 130.383307 1.297699 0.241880 0.284156 -0.617877 0.828947 0.960014 P
0.156584 0.136055 6.868240 3.955145 0.982800 0.935869 13.736479 0.954496 179.063612 1.000095 0.119269 0.171095 -0.759555 0.840987 0.976521 P
0.084779 0.281854 6.044021 4.798936 0.970634 0.875275 12.088043 1.121745 139.173584 1.220143 0.216909 0.262158 -0.644473 0.829067 0.963457 P
0.133734 0.237987 6.692590 5.875291 0.979747 0.894807 13.385181 1.063958 175.075705 1.146158 0.191772 0.238155 -0.679935 0.832759 0.971329 P
0.078491 0.292862 6.267931 4.989716 0.970653 0.865747 12.535862 1.145128 149.414822 1.244695 0.218572 0.268387 -0.633139 0.827614 0.964347 P
0.100397 0.244372 6.404288 5.091834 0.976004 0.888809 12.808576 1.095390 157.321690 1.179019 0.193276 0.243214 -0.669690 0.833993 0.970119 P
0.092429 0.262963 6.114990 4.855092 0.972919 0.880597 12.229980 1.095819 143.126963 1.186112 0.203998 0.253326 -0.653348 0.827395 0.968014 P
0.098441 0.240429 6.164085 4.704291 0.974446 0.890807 12.328169 1.079979 145.098515 1.162934 0.191121 0.240972 -0.669801 0.831285 0.970858 P
0.096299 0.228948 6.164995 4.664114 0.975456 0.893870 12.329991 1.086886 144.834907 1.164661 0.182707 0.235039 -0.678815 0.835801 0.970154 P
0.114918 0.190667 7.044179 4.611532 0.979327 0.910719 14.088359 1.058236 188.039563 1.122324 0.158059 0.211430 -0.709683 0.842314 0.974295 P

0.145978 0.207012 6.843536 4.217347 0.975457 0.905175 13.687071 0.997193 177.695378 1.068085 0.169940 0.221173 -0.686061 0.820326 0.968792 P
0.070599 0.320734 6.230212 5.010925 0.967997 0.856165 12.460424 1.159213 147.440348 1.269668 0.234781 0.280785 -0.616352 0.823538 0.965066 P
0.077644 0.310375 6.131844 4.575741 0.966085 0.856477 12.263688 1.138666 141.758691 1.243473 0.225733 0.276197 -0.610919 0.816039 0.964070 P
0.105141 0.252166 5.854728 4.409819 0.971409 0.884325 11.709455 1.049233 131.027449 1.135514 0.197027 0.247909 -0.649116 0.815640 0.975184 P
0.130453 0.222929 6.036391 5.011711 0.977759 0.896071 12.072781 1.001754 142.391558 1.076859 0.178674 0.231843 -0.676061 0.817361 0.982351 P
0.093285 0.241622 5.826490 4.659726 0.974073 0.888735 11.652980 1.080624 130.171990 1.163040 0.190677 0.242103 -0.666932 0.829838 0.973470 P
0.112793 0.213396 5.933137 4.901811 0.978233 0.901284 11.866274 1.035041 136.709454 1.107808 0.173360 0.225659 -0.689381 0.830014 0.980661 P
0.100552 0.233195 5.822806 4.727546 0.975337 0.892609 11.645612 1.063971 130.648118 1.143592 0.185737 0.237356 -0.672868 0.829003 0.976490 P
0.088742 0.229131 5.399870 6.576532 0.982580 0.893370 10.799740 1.129144 119.597432 1.206113 0.182516 0.235394 -0.697742 0.851900 0.975807 P
0.069994 0.287905 5.822106 6.814348 0.978875 0.867727 11.644212 1.189505 136.270379 1.286697 0.215846 0.266104 -0.657661 0.846652 0.972527 P
0.093365 0.204508 5.570816 5.813012 0.982409 0.903469 11.141632 1.096789 123.946410 1.164874 0.166495 0.220808 -0.708878 0.849887 0.978282 P
0.115063 0.213428 6.079649 6.483938 0.983542 0.902843 12.159298 1.066854 148.764632 1.139639 0.174423 0.225016 -0.704090 0.843130 0.977907 P
0.095887 0.195692 5.524844 6.577849 0.985125 0.908858 11.049687 1.098493 125.141979 1.164561 0.161645 0.214636 -0.726708 0.857640 0.980247 P
0.087132 0.228299 5.512939 6.093192 0.981266 0.896403 11.025878 1.110473 122.459710 1.188723 0.183901 0.233943 -0.696304 0.848205 0.976419 P
0.124906 0.177041 6.029174 6.456137 0.986289 0.915997 12.058347 1.033708 147.190183 1.092207 0.148307 0.202760 -0.730451 0.846302 0.982800 P
0.097373 0.221355 6.105444 3.930460 0.971841 0.893707 12.210888 1.061490 139.809551 1.133997 0.175538 0.231350 -0.672627 0.826938 0.968868 P
0.114634 0.194558 6.377286 4.543658 0.978590 0.908950 12.754572 1.036945 155.282855 1.102704 0.160637 0.214106 -0.709585 0.838409 0.976128 P
0.125821 0.185801 6.454558 4.693962 0.980209 0.911326 12.909116 1.020306 159.933849 1.081683 0.153847 0.208974 -0.713915 0.836324 0.976949 P
0.119081 0.178217 6.257061 4.825237 0.981533 0.915174 12.514121 1.025442 151.112474 1.084482 0.148948 0.203642 -0.723852 0.841412 0.980085 P
0.094566 0.201967 5.473063 4.073900 0.975212 0.902216 10.946126 1.068605 113.659051 1.134047 0.163302 0.219702 -0.693655 0.836833 0.974692 P
0.253601 0.106398 7.039190 4.623508 0.988494 0.948559 14.078379 0.825614 194.023416 0.860265 0.095693 0.146363 -0.787077 0.820548 0.986944 P
0.099761 0.226024 6.131713 4.379406 0.974195 0.896698 12.263426 1.077282 142.421425 1.154618 0.181991 0.232866 -0.677975 0.833510 0.966367 P
0.103081 0.173955 5.346331 3.850278 0.977410 0.916047 10.692661 1.053920 108.132490 1.110661 0.145423 0.201042 -0.719492 0.844868 0.975056 P
0.095230 0.199281 5.710874 4.835710 0.979395 0.907027 11.421748 1.072078 126.259242 1.139595 0.163874 0.217063 -0.708178 0.844794 0.977771 P
0.095775 0.227190 5.194104 3.927546 0.971077 0.892477 10.388207 1.070962 102.294047 1.146585 0.180070 0.234614 -0.666862 0.826323 0.966528 P
0.086693 0.231988 5.586482 4.569128 0.974614 0.890948 11.172963 1.094046 119.629106 1.171742 0.183404 0.237248 -0.676086 0.835450 0.972387 P
0.085508 0.246659 5.452456 3.869069 0.968124 0.883355 10.904911 1.097874 111.407598 1.180123 0.191190 0.245408 -0.653859 0.826157 0.963268 P
0.082573 0.268704 5.001768 4.023230 0.966606 0.873570 10.003536 1.107016 94.972288 1.196818 0.203424 0.256824 -0.636392 0.820604 0.959911 P
0.089890 0.225947 5.668048 4.763076 0.976281 0.894277 11.336097 1.086808 123.874288 1.162979 0.180210 0.233645 -0.685124 0.838311 0.977130 P
0.093549 0.195945 5.953552 3.980949 0.975390 0.906549 11.907103 1.074144 133.080869 1.139142 0.160447 0.215646 -0.701362 0.841750 0.969999 P
0.101557 0.185751 5.528848 3.970322 0.976608 0.909535 11.057695 1.059903 115.651384 1.119513 0.152724 0.209282 -0.704931 0.839611 0.968999 P
0.080013 0.256020 4.807953 4.867149 0.973699 0.881367 9.615906 1.134291 91.150355 1.220965 0.198232 0.250227 -0.663628 0.838318 0.966039 P
0.106731 0.198443 5.577993 3.377579 0.970623 0.904369 11.155986 1.025042 115.947898 1.089834 0.161403 0.217454 -0.682384 0.823076 0.963781 P
0.112078 0.151371 5.763767 3.315541 0.977172 0.925986 11.527533 1.027598 123.359437 1.075885 0.129294 0.184772 -0.736241 0.845765 0.970208 P
0.104564 0.205876 5.658373 3.461230 0.970260 0.900778 11.316746 1.047408 119.098348 1.114580 0.166003 0.222106 -0.677801 0.825704 0.965284 P
0.099076 0.186177 5.559657 3.868549 0.975937 0.909661 11.119315 1.066713 116.342812 1.126852 0.153200 0.209532 -0.705899 0.841645 0.969640 P

A.2 TRAINING AND TEST *.dat FILES

A *.dat file contains columns of data where each column corresponds to the values of a chosen property and each row corresponds to one data point. The last column corresponds to the class or category of each data point as given in the corresponding rows. In this study, the possible values for the last column are: 1–normal, 0–cancerous, and 0.5–adenomatous polyp. Below is a print-out of the *.dat training data for the feature set consisting of Mean, Sum Average and Sum Variance (Set A).

"trainDataANFIS_mean_sumAve_sumVar.dat" – DAT file containing the properties mean, sum average, and sum variance:

3.854912 7.709824 55.697310 1
3.405950 6.811900 42.964237 1
4.347363 8.694727 78.654562 1
3.739664 7.479329 57.300597 1
4.299960 8.599920 67.620411 1
4.014439 8.028878 59.556406 1
3.441644 6.883288 43.200620 1
4.581609 9.163219 74.511452 1
4.544380 9.088760 73.179009 1
4.824253 9.648506 84.001432 1
4.340859 8.681719 66.981169 1
4.086981 8.173963 61.463214 1
4.513994 9.027987 73.546979 1
3.841698 7.683397 53.086178 1
4.393425 8.786849 68.467952 1
4.280154 8.560308 64.531489 1
4.094082 8.188163 59.053699 1
4.723782 9.447564 84.607172 1
4.256759 8.513518 65.084140 1
4.997390 9.994781 89.961689 1
5.064641 10.129281 91.969272 1
4.434146 8.868292 69.561133 1
4.793410 9.586820 85.728799 1
4.758752 9.517505 82.177521 1
4.667151 9.334303 78.807144 1
5.068178 10.136355 91.734812 1
4.820580 9.641160 83.982859 1
4.981734 9.963468 88.440245 1
5.002666 10.005333 93.034964 1
4.919476 9.838951 94.365043 1
4.172789 8.345578 68.247214 1
4.600005 9.200010 81.060261 1
5.091449 10.182898 94.484210 1
4.754144 9.508288 83.515864 1
4.838808 9.677617 89.456490 1
5.041280 10.082560 96.100107 1
5.760539 11.521077 125.200552 1
5.821129 11.642257 126.642359 1
5.844854 11.689709 130.391069 1
5.571689 11.143377 119.085726 1

5.676531 11.353062 120.308759 1
5.020085 10.040169 95.208169 1
5.420305 10.840610 108.026179 1
4.947813 9.895627 92.412707 1
4.922006 9.844013 89.534139 1
4.874427 9.748855 87.437906 1
4.930548 9.861097 87.882763 1
5.090248 10.180496 94.471550 1
5.218010 10.436019 100.084188 1
4.856024 9.712049 87.831055 1
4.270760 8.541520 66.538241 1
5.027149 10.054299 91.337398 1
5.179615 10.359229 100.749176 1
4.451664 8.903328 69.083051 1
4.872199 9.744399 85.509462 1
4.615352 9.230705 76.489155 1
3.963760 7.927520 60.411368 1
4.065005 8.130010 61.091971 1
4.008447 8.016894 60.985401 1
4.277662 8.555325 70.110233 1
4.324001 8.648003 73.569866 1
4.549430 9.098860 80.621538 1
3.906520 7.813040 54.744094 1
4.385161 8.770322 69.587402 1
4.381332 8.762665 68.804701 1
4.480200 8.960399 73.036422 1
4.488659 8.977318 76.665877 1
4.304512 8.609023 67.414519 1
4.075295 8.150589 61.367574 1
4.252149 8.504298 66.277780 1
7.183031 14.366061 191.260491 0
7.636479 15.272957 218.530056 0
7.167858 14.335715 188.069580 0
6.966724 13.933448 178.390091 0
7.128353 14.256706 188.489619 0
7.271751 14.543501 196.585620 0
7.381265 14.762529 201.896502 0
7.060977 14.121953 186.273261 0
7.220471 14.440942 198.626582 0
7.037086 14.074172 186.749197 0
7.539689 15.079378 212.457854 0
7.155872 14.311743 188.908436 0
6.896017 13.792034 179.147877 0
7.627888 15.255775 217.983309 0
7.303785 14.607570 202.174431 0
7.696248 15.392495 225.898757 0
7.030759 14.061518 187.365736 0
6.954381 13.908762 184.558821 0
7.035553 14.071105 188.456420 0
6.999354 13.998707 185.952898 0
6.988945 13.977890 186.720196 0
6.332546 12.665092 149.941575 0
6.476329 12.952657 156.393449 0
6.931258 13.862516 178.013281 0
6.089851 12.179701 141.196755 0
6.503541 13.007083 159.743004 0
6.160106 12.320211 139.887767 0
6.943266 13.886533 182.751474 0
7.241440 14.482881 194.072151 0
5.806192 11.612383 122.632883 0
6.344025 12.688050 146.674718 0
6.570614 13.141227 158.846436 0
6.489143 12.978286 154.818452 0
6.467621 12.935242 152.881610 0

6.794233 13.588466 170.121545 0
9.699039 19.398077 359.522313 0
9.701309 19.402619 357.118623 0
9.626044 19.252088 351.435946 0
9.331532 18.663064 332.331355 0
9.617511 19.235021 348.367744 0
9.407274 18.814549 333.563460 0
9.653455 19.306910 347.307314 0
8.525912 17.051824 292.405557 0
8.834425 17.668850 313.031259 0
8.913820 17.827640 322.277579 0
8.635229 17.270458 306.191466 0
8.611913 17.223826 310.292134 0
9.349980 18.699960 351.731527 0
8.469025 16.938049 291.029486 0
8.724471 17.448941 311.878280 0
9.750931 19.501861 356.824059 0
9.613230 19.226460 346.874029 0
9.779795 19.559590 359.619443 0
9.488428 18.976856 337.698308 0
9.674387 19.348774 356.072077 0
9.767235 19.534470 363.092691 0
9.609018 19.218037 348.133652 0
9.742681 19.485361 358.251753 0
9.854520 19.709039 370.620828 0
9.609084 19.218167 351.844741 0
9.544894 19.089788 344.480738 0
9.851194 19.702388 366.182926 0
9.693315 19.386630 354.821390 0
10.171430 20.342860 389.040213 0
9.817274 19.634548 357.231305 0
9.637890 19.275781 348.220864 0
9.455861 18.911722 339.656753 0
9.152853 18.305705 319.225344 0
9.254700 18.509401 329.437033 0
9.232246 18.464492 326.433775 0
4.522802 9.045605 73.254333 0.5
4.444204 8.888407 70.655545 0.5
4.625903 9.251806 76.822889 0.5
4.494544 8.989088 72.772451 0.5
4.602262 9.204524 76.842878 0.5
4.991412 9.982825 90.939519 0.5
4.579155 9.158311 76.469769 0.5
4.939416 9.878832 102.246833 0.5
5.467365 10.934729 120.826589 0.5
5.513248 11.026495 119.393910 0.5
5.005086 10.010171 107.883590 0.5
5.161304 10.322609 113.358825 0.5
5.038429 10.076858 109.115351 0.5
5.489082 10.978163 117.273202 0.5
5.668540 11.337080 119.100763 0.5
5.905068 11.810137 130.595280 0.5
6.262556 12.525111 147.214753 0.5
5.969540 11.939081 131.848859 0.5
6.142597 12.285194 140.071817 0.5
6.430478 12.860956 153.584139 0.5
5.496569 10.993138 111.313881 0.5
5.651354 11.302708 122.192313 0.5
5.645619 11.291238 122.831914 0.5
6.621852 13.243705 164.984331 0.5
5.843587 11.687174 127.929516 0.5
6.865811 13.731622 176.549425 0.5
6.042852 12.085704 136.694338 0.5
6.690660 13.381320 172.455251 0.5

6.267984 12.535967 147.003561 0.5
6.402081 12.804163 154.751474 0.5
6.112728 12.225455 140.721912 0.5
6.162209 12.324418 142.800006 0.5
6.162899 12.325798 142.445012 0.5
7.043324 14.086648 185.478201 0.5
6.840946 13.681893 175.505006 0.5
6.232928 12.465855 145.307630 0.5
6.134137 12.268274 139.653010 0.5
5.854230 11.708459 129.078037 0.5
6.036831 12.073662 140.578749 0.5
5.826353 11.652706 128.031916 0.5
5.934614 11.869229 134.796913 0.5
5.823793 11.647587 128.586371 0.5
5.395715 10.791430 117.120353 0.5
5.826219 11.652438 133.912359 0.5
5.569147 11.138294 121.641080 0.5
6.081793 12.163586 146.338216 0.5
5.524392 11.048783 122.748025 0.5
5.512669 11.025338 119.922023 0.5
6.031815 12.063630 144.954580 0.5
6.101590 12.203180 137.677267 0.5
6.377018 12.754037 153.065551 0.5
6.453486 12.906972 157.817981 0.5
6.256852 12.513705 148.968660 0.5
5.471800 10.943601 111.748908 0.5
7.038345 14.076689 192.358674 0.5
6.133179 12.266358 140.001196 0.5
5.349048 10.698097 106.207090 0.5
5.712694 11.425389 124.196763 0.5
5.194634 10.389268 100.250455 0.5
5.586711 11.173422 117.536209 0.5
5.454295 10.908590 109.514880 0.5
5.003708 10.007415 92.868517 0.5
5.668782 11.337565 121.861446 0.5
5.953042 11.906083 130.775130 0.5
5.526730 11.053461 113.342333 0.5
4.808131 9.616262 88.805774 0.5
5.578828 11.157656 113.772120 0.5
5.765997 11.531994 121.162450 0.5
5.652374 11.304748 116.733798 0.5
5.559787 11.119574 114.119319 0.5

A.3 MATLAB program "image2FeatureDATAFile.m" used to calculate the textural properties of each training image and store in a .data file:

```
%This program calculates the textural properties of images located in the
%"fileLocation" directory. The output is a data set written on a .data
%text file in the current directory of MATLAB. There are a number of
%important settings that have to be made prior to execution of this
%program. To avoid execution errors and errors in the output data set:
%   1. Set the quantization levels in computing GLCM through
%      "quantizationForTexture".
%   2. Make sure that "fileLocation" is assigned the correct location of images
%   3. The images in the "fileLocation" can either be RGB or GRAYSCALE only
%   4. There should be no other files besides the subject images in the
%         "fileLocation" directory
%   5. Be sure to specify the name of the .data file in the
%         "filenameDATAfile"
%   6. Be sure to specify the scale of image resize [1.0, 0.75, etc...]
%   7. In the for-loop, specify the necessary commands to be executed to
%         each image before calculating the textural properties. Do it
%         for both RGB and GRAYSCALE images [imageType == 1 & 2].
%   8. During execution, the user will be shown a list of file entries after
%         executing the "dir" command and will be asked as to where the
%         list of the 'actual' image files begin. Usually, the first
%         entry is just a dot(.) then followed by two dots (..) and then
%         comes the first image file *.jpg. In this case, one must key in
%         "3" since the first image file is at the 3rd entry in the dir.
%
%%A peculiar thing about Notepad text editor is that it cannot seem to be
%%able correctly interpret the newline escape sequence '\n'. Instead of
%%the usual new line, Notepad displays '\n' as something like an 'o'.
%%Putting the carriage return escape sequence '\r' before '\n' seems to
%%solve the problem. Therefore, for data to be readable once opened by
%%Notepad, use '\r\n' for new line instead of just '\n'.
%
%
%AUTHOR:
%Laurence A. Gan Lim
%Research Student
%BIOCORE, Faculty of Engineering and Computing
%Coventry University
%
%updated: 9 Aug 2011, 22:34 for the pathologist survey images

%IMPORTANT SETTINGS BEFORE PROGRAM EXECUTION:
%start = 3; %Assumes that the list of filenames starts at the ith entry when dir() is executed.
%'fileLocation' is where the image files are stored. There should be NO other
%files in this directory to avoid execution errors.

%quantizationForTexture = 32;
%quantizationForTexture = 24;
quantizationForTexture = 16;
%quantizationForTexture = 8;

fileLocation = '{specify location of images here}';

imageType = input('enter the image type at source:    [ 1 for RGB    2 for GRAYSCALE]\n');
if imageType ~= 1 && imageType ~= 2
   error('invalid type of image')
end



%DESTINATION OF TEXT FILE IS THE CURRENT DIRECTORY:
filenameDATAfile = 'surveyTestImages.data'; scale = 0.25; % 25% of the original
```

```
%All files above where generated using quantizationForTexture = 16;


dir(fileLocation)
fprintf('....processing to produce %s....\n\n', filenameDATAfile)
start = input('start of images from dir:        [press ZERO to abort]\n');
if start == 0
   error('program execution aborted')
end

fprintf('\n....%i files to be processed....\n',size(dir(fileLocation),1)-start+1)

tic % timer start



%PUT NAMES OF TEXTURE PROPERTIES IN A CELL ARRAY OF STRINGS....
featureVectorLength = 15; %number of components or dimensions
componentNames = cell(featureVectorLength,1); %initialize cell array of strings for component names
componentNames{1} = 'ASM';
componentNames{2} = 'contrast';
componentNames{3} = 'mean';
componentNames{4} = 'variance';
componentNames{5} = 'correlation';
componentNames{6} = 'IDM';
componentNames{7} = 'sumAverage';
componentNames{8} = 'sumEntropy';
componentNames{9} = 'sumVariance';
componentNames{10} = 'entropy';
componentNames{11} = 'differenceVariance';
componentNames{12} = 'differenceEntropy';
componentNames{13} = 'IMC12';
componentNames{14} = 'IMC13';
componentNames{15} = 'MCC';

%PREPARE .DATA FILE FOR WRITING....
fid = fopen(filenameDATAfile,'w');
if fid == -1
   error('cannot open file file for writing');
end

%WRITE THE 'HEADER INFO' TO THE FILE...
fprintf(fid,'%i',featureVectorLength); % 1st entry: number indicating no. of dimensions
fprintf(fid,'\r\n#l image classification');
fprintf(fid,'\r\n## N - normal, P - adenomatous polyp, C - cancerous');
fprintf(fid,'\r\n#n');
for a = 1:featureVectorLength
   fprintf(fid,' %s',componentNames{a}); % '#n ASM contrast mean .....'
end
fprintf(fid,'\r\n');


%THE FOR-LOOP BELOW READS THE IMAGE FILES IN THE CHOSEN DIRECTORY IN THE
%fileLocation AND WRITES THE CORRESPONDING GLCM TEXTURE PROPERTIES IN THE
%.DATA FILE POINTED TO BY THE fid, THE FILE IDENTIFIER NUMBER.
files = dir(fileLocation);
fnames = fieldnames(files);
fprintf('percent completed:\n')
for a = start:size(files,1)
   fileName = getfield(files,{a},fnames{1});
   %PERFORM CHECK OF 'VALID' IMAGE FILENAMES
   if fileName(1) == 'n' classification = 'N';
   elseif fileName(1) == 'p' classification = 'P';
   elseif fileName(1) == 'c' classification = 'C';
```

```
    else
        fclose(fid);
        error('filename of image does not start with n, p, or c');
    end
    fileName = strcat(fileLocation,fileName);
    image = imread(fileName); %now, read the image into memory
    if imageType == 1 % 1 for RGB at source
        image = imresize(rgb2gray(image),scale);
        %image = histeq(imresize(rgb2gray(image),scale)); %with histogram equalization
        %f = fspecial('unsharp',0.5);    %create the spatial filter [unsharp masking]
        %image = uint8(filter2(f,imresize(rgb2gray(image),scale))); %apply the filter
    end
    if imageType == 2 % 1 for GRAYSCALE at source
        image = imresize(image,scale);
        %image = histeq(imresize(image,scale)); %with histogram equalization
        %f = fspecial('unsharp',0.5);    %create the spatial filter [unsharp masking]
        %image = uint8(filter2(f,imresize(image,scale))); %apply the filter
    end
    tprop = glf_glcmTexture(image,quantizationForTexture,256); %calculate the properties
    data = struct2cell(tprop);
    for b = 1:featureVectorLength
        fprintf(fid,'%f ',data{b}); %write the properties into a .data file
    end
    fprintf(fid,'%s\r\n',classification);
    clear tprop;
    %DISPLAY PERCENT COMPLETED:
    if mod(100*a/(size(files,1)-start+1),10) == 0
        fprintf('%i ',(a/(size(files,1)-start+1))*100)
    end
end

fprintf('\nYou can now view the .data file in the current directory to see the results.\n')

fclose(fid);

clear

toc %report timer elapsed time
```

A.4 MATLAB function "glf_computeVarianceRatio.m" used to calculate the variance ratio of each textural property for the entire training image set:

```
function vr = glf_computeVarianceRatio(data, category)
%Function "glf_computeVarianceRatio()" calculates the variance ratios of a
%given data set corresponding to the given components. The input argument
%"data" is assumed to be a matrix of data vectors in column format - each
%column is a data point while each row is a set of values for a single
%vector component. The input argument "category" is assumed to be a row
%vector of characters with each character representing the category
%corresponding to a particular column in the "data" matrix. The entire
%function returns a column vector containing the variance ratio
%corresponding to each component in the given "data" matrix which was
%assumed to be in column format.
%
%THEORETICAL BASIS:
%Modified version of Multiple Discriminant Analysis Criterion (Duda and
%Hart, 1973 and 2001). Used by Boland et al. (1998) and Atlamazolgou et al.
%(2001). FOR EACH FEATURE:
%
% variance ratio = variance of feature using all samples /
%                  sum of variances of same feature per class
%
%Feature with large variance ratio means good feature for classification.
%
%
%EXAMPLE:
%trainData = glf_readTrainOrTestDATAfile('trainingData.data');
%trainData.componentNames %display component names
%ratio = glf_computeVarianceRatio(trainData.data,trainData.category)
%barh([1:15], ratio) %MATLAB bar graph does not accept strings as axis values
%                  %BETTER TO USE MS EXCEL BAR GRAPH IN THIS CASE
%
%AUTHOR:
%Laurence A. Gan Lim
%Research Student
%BIOCORE, Faculty of Engineering and Computing
%Coventry University
%
%last updated:
%21:37 May 19, 2011

%PERFORM BASIC CHECKING OF INPUT ARGUMENTS
if size(data,2) ~= size(category,2)
   error('input arguments must have equal number of columns')
end

%THE FIRST ORDER OF BUSINESS IS TO EXAMINE THE NUMBER OF DISTINCT
%CATEGORIES THAT WERE GIVEN IN THE INPUT ARGUMENT:
class = category(1);
for a = 1:size(category,2)
   template = repmat(category(a),1,size(class,2));
   total = sum(template==class);
   if total == 0
      class(size(class,2)+1) = category(a);
   end
   clear template total
end
%Now, the variable "class" holds the given distinct categories.


%GET THE LOCATIONS OF VECTORS IN THE DATA BELONGING TO EACH CATEGORY
index = cell(size(class,2),1);
for a = 1:size(class,2)
```

```
    inspector = repmat(class(a),1,size(category,2));
    %strcat('index',num2str(a)) = find(inspector==category)
    index{a} = find(inspector==category);
end
%Now, the cell "index{}" holds the indices of the vectors under each class.
%class(1) corresponds to index{1}, class(2) to index{2}, .... and so on...

%COMPUTE NOW THE BETWEEN-CLASS VARIANCE - ALL SAMPLES CONSIDERED FOR EACH
PROPERTY OR COMPONENT
btcVariance = var(data,0,2);
%btcVariance is a column matrix. The command above applies the var command
%on a per row basis on the input variable "data".


%COMPUTE NOW THE WITHIN-CLASS VARIANCE - ONE FOR EACH CLASS
winVariance = zeros(size(data,1),size(class,2));
for a = 1:size(class,2)
    winVariance(:,a) = var(data(:,index{a}),0,2);
end

sumWinVar = sum(winVariance,2);

vr = btcVariance./sumWinVar; %returns the variance ratios as a column vector
end
```

A.5 MATLAB function "glf_SOMFitnessFunction.m" as the fitness function used in the MATLAB GA Toolbox

```
function somQuality = glf_SOMFitnessFunction(inputRowVector)
%glf_SOMFitnessFunction(inputRowVector) is a function that is used by the
%genetic algorithm toolbox to implement feature selection using the
%somQuality as fitness function. KSOM is implemented using a SOM Toolbox
%from HUT. The input argument "inputRowVector" accepts a row of numbers
%used as coefficients of the texture property values read from
%"trainingData.data". It is hoped that the absolute values of the
%coefficients are indicative of the good discriminating characteristics
%of the different texture properties. IMPORTANT: MATLAB GA TOOLBOX
%OPTIMIZES BY MINIMIZING THE FITNESS FUNCTION. The entire function returns
%qe, which is the average quantization error of the map, as the parameter
%to be minimized by the GA Toolbox.
%
%To use this and the GA Toolbox:
%- Click START of MATLAB, go to Toolboxes and find "gatool"
%- type "@glf_SOMFitnessFunction" on the Fitness Function
%- enter 15 as number of variables
%- population size: 20(default), 30-takes longer but may be more effective
%- Bounds: lower 0      upper ___
%- find "Stopping criteria" and indicate 15 generations
%- find "Plot functions" and check "Best fitness"
%- accept the other default settings
%- make sure to execute "clear,clc" before going further
%- click START button of the Optimization Tool
%
%AUTHOR:
%Laurence A. Gan Lim
%Research Student
%BIOCORE, Faculty of Engineering and Computing
%Coventry University
%
%last updated:
%15:53 May 21, 2011



%load the data
sD = som_read_data('trainingData300x400standardised.data');

if size(inputRowVector,2) ~= size(sD.data,2)
   error('incompatible given inputRowVector');
end


inputRowVector = repmat(inputRowVector, size(sD.data,1),1);
sD.data = (sD.data).*inputRowVector;

sD = som_normalize(sD, 'var');

%make the SOM
sM = som_make(sD,'munits',200);
sM = som_autolabel(sM,sD,'vote');

%basic visualization [problem with handling so many components]
%som_show(sM,'umat','all','comp',1:15, 'empty','Labels','norm','d');
%som_show_add('label',sM,'subplot',17);


%And now, some quantitative analysis of SOM:
%where...
```

```
% qe --> average quantization error - simply the ave. distance (weighted
%                               with the mask) from each data vector
%                               to its BMU
% te --> topographic error - gives the percentage of data vectors for
%                       which the BMU and the second-BMU are not
%                       neighboring map units (Kimmo Kiviluoto, 1996)
[qe, te] = som_quality(sM,sD);

%qe = qe/0.5;
%te = te/0.01;


%QUANTITY TO BE MINIMIZED BY MATLAB GA TOOLBOX
x = var(inputRowVector(1,:))/var([1 0 1 0 1 0 1 0 1 0 1 0 1 0 1]);
k = 0.1;
y = k*(1-x)/(k+x);
somQuality = qe + y;
```

A.6 MATLAB program "writeToFileChosenPropertiesForANFIS.m" to generate the training and test *.dat files from the *.data files.

*[The main difference between a *.dat file and a *.data file is that a *.dat file only contains the properties that were selected in the feature selection process.]*

```
%Save chosen texture properties into a .dat file for ANFIS.
%
%Make sure you do the following before running this program:
%- choose the properties you want to write into the .DAT file.
%- specify the filenames of the .DATA training and testing files where data
%   will be read
%- specify the filenames of the .DAT training and testing files where data
%   will be saved
%- ALL FILES WILL BE SAVED AT THE CURRENT DIRECTORY!
%
%AUTHOR:
%Laurence A. Gan Lim
%Research Student
%BIOCORE, Faculty of Engineering and Computing
%Coventry University
%
%updated:
%   2 June 2011, 00:31

labels = zeros(15,1);

%chosen property below is = 1
% [ WARNING: >4 inputs can cause ANFIS to crash ]
label(1) = 0; %ASM
label(2) = 1; %contrast
label(3) = 0; %mean
label(4) = 0; %variance
label(5) = 0; %correlation
label(6) = 1; %IDM
label(7) = 0; %sumAverage
label(8) = 0; %sumEntropy
label(9) = 0; %sumVariance
label(10) = 0; %entropy
label(11) = 1; %differenceVariance
label(12) = 0; %differenceEntropy
label(13) = 0; %IMC12
label(14) = 0; %IMC13
label(15) = 0; %MCC

%for TRAINING DATA....
fid1 = fopen('trainingData300x400.data','r');

%fid2 = fopen('trainDataANFIS_mean_sumAve_sumVar.dat','w');
%fid2 = fopen('trainDataANFIS_mean_sumAve.dat','w');
%fid2 = fopen('trainDataANFIS_variance_corr.dat','w');
%fid2 = fopen('trainDataANFIS_mean_IDM_DE.dat','w');
%fid2 = fopen('trainDataANFIS_contrast_sumV_imc12.dat','w');
%fid2 = fopen('trainDataANFIS_contrast_entropy_diffVar.dat','w'); %set C
%fid2 = fopen('trainDataANFIS_contrast_IDM_sumVar.dat','w'); %set D
%fid2 = fopen('trainDataANFIS_sumAve_diffEntropy.dat','w'); %set E
fid2 = fopen('trainDataANFIS_contrast_IDM_diffVar.dat','w'); % for set F


%for TESTING DATA....
fid3 = fopen('testData300x400.data','r');
```

```
%fid4 = fopen('testDataANFIS_mean_sumAve_sumVar.dat','w');
%fid4 = fopen('testDataANFIS_mean_sumAve.dat','w');
%fid4 = fopen('testDataANFIS_variance_corr.dat','w');
%fid4 = fopen('testDataANFIS_mean_IDM_DE.dat','w');
%fid4 = fopen('testDataANFIS_contrast_sumV_imc12.dat','w');
%fid4 = fopen('testDataANFIS_contrast_entropy_diffVar.dat','w'); %set C
%fid4 = fopen('testDataANFIS_contrast_IDM_sumVar.dat','w'); %set D
%fid4 = fopen('testDataANFIS_sumAve_diffEntropy.dat','w'); %set E
fid4 = fopen('testDataANFIS_contrast_IDM_diffVar.dat','w'); %set F

%======================== for TRAINING DATA.... ==========================
numberOfSamples = 210;

%deal with the first line of data....
fseek(fid1,225,0);
for a = 1:15
   string = fscanf(fid1,'%s',1);
   if label(a) ~= 0
      fprintf(fid2,'%s ',string);
   end
end
string = fscanf(fid1,'%s',1);
if string == 'N'
   fprintf(fid2,'0');
elseif string == 'P'
   fprintf(fid2,'0.5');
elseif string == 'C'
   fprintf(fid2,'1');
else
   error('unusual classification in the source file');
   fclose(fid1);
   fclose(fid2);
end

%then, deal with the 2nd line of data until the end of file....
for j = 2:numberOfSamples
   fprintf(fid2,'\r\n');
   for a = 1:15
      string = fscanf(fid1,'%s',1);
      if label(a) ~= 0
         fprintf(fid2,'%s ',string);
      end
   end
   string = fscanf(fid1,'%s',1);
   if string == 'N'
      fprintf(fid2,'0');
   elseif string == 'P'
      fprintf(fid2,'0.5');
   elseif string == 'C'
      fprintf(fid2,'1');
   else
      error('unusual classification in the source file');
      fclose(fid1);
      fclose(fid2);
   end
end

fclose(fid1);
fclose(fid2);


%======================== for TESTING DATA....==========================

numberOfSamples = 90;
```

```
%deal with the first line of data....
fseek(fid3,225,0);
for a = 1:15
    string = fscanf(fid3,'%s',1);
    if label(a) ~= 0
        fprintf(fid4,'%s ',string);
    end
end
string = fscanf(fid3,'%s',1);
if string == 'N'
    fprintf(fid4,'0');
elseif string == 'P'
    fprintf(fid4,'0.5');
elseif string == 'C'
    fprintf(fid4,'1');
else
    error('unusual classification in the source file');
    fclose(fid3);
    fclose(fid4);
end

%then, deal with the 2nd line of data until the end of file....
for j = 2:numberOfSamples
    fprintf(fid4,'\r\n');
    for a = 1:15
        string = fscanf(fid3,'%s',1);
        if label(a) ~= 0
            fprintf(fid4,'%s ',string);
        end
    end
    string = fscanf(fid3,'%s',1);
    if string == 'N'
        fprintf(fid4,'0');
    elseif string == 'P'
        fprintf(fid4,'0.5');
    elseif string == 'C'
        fprintf(fid4,'1');
    else
        error('unusual classification in the source file');
        fclose(fid3);
        fclose(fid4);
    end
end

fclose(fid3);
fclose(fid4);

clear
```

A.7 MATLAB program "ANFISthesisImplementationCommandLine.m" to implement ANFIS and produce the necessary classification results.

```
%Implementation of ANFIS for the training data and testing data....
%
%Make sure to do the following before running this program:
%- specify the location and file name of the file where the variables will
%   be saved after program execution. Better to create a new folder.
%- specify the training and testing .dat filenames for the 'load' commands below
%- save the graphs
%- copy the program execution messages and save in a text file for future
%   reference
%
%AUTHOR:
%Laurence A. Gan Lim
%Research Student
%BIOCORE, Faculty of Engineering and Computing
%Coventry University
%
%updated:
%   26 May 2011, 23:25
%   27, May 2011 20:12
%   30 May 2011, 23:14
%   2 June 2011, 00:15
%   28 July 2011, 22:25
%   30 July 2011, 00:03



%SPECIFY LOCATION WHERE THE FINAL VALUES OF THE USED VARIABLES WILL BE SAVED
%fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run A\';
%fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run B\';
%fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run C\';
%fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run D\';
%fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run E\';
fileSaveLocation = 'C:\Users\lagrange\GAN LIM 03\PhD work\dissertation v3\ANFIS run F\';

%SPECIFY ALSO THE FILE NAME OF THE FILE WHERE THE VARIABLES WILL BE STORED
%fileSaveName = 'variablesANFISa.mat';
%fileSaveName = 'variablesANFISb.mat';
%fileSaveName = 'variablesANFISc.mat';
%fileSaveName = 'variablesANFISd.mat';
%fileSaveName = 'variablesANFISe.mat';
fileSaveName = 'variablesANFISf.mat';


%SPECIFY THE FILE NAMES OF THE .DAT FILES TO BE USED BY ANFIS:
%set A:
%trainingDataANFIS = load('trainDataANFIS_mean_sumAve_sumVar.dat');
%testingDataANFIS = load('testDataANFIS_mean_sumAve_sumVar.dat');
%set B:
%trainingDataANFIS = load('trainDataANFIS_mean_sumAve.dat');
%testingDataANFIS = load('testDataANFIS_mean_sumAve.dat');
%set C:
%trainingDataANFIS = load('trainDataANFIS_contrast_entropy_diffVar.dat');
%testingDataANFIS = load('testDataANFIS_contrast_entropy_diffVar.dat');
%set D:
%trainingDataANFIS = load('trainDataANFIS_contrast_IDM_sumVar.dat');
%testingDataANFIS = load('testDataANFIS_contrast_IDM_sumVar.dat');
%set E:
%trainingDataANFIS = load('trainDataANFIS_sumAve_diffEntropy.dat');
%testingDataANFIS = load('testDataANFIS_sumAve_diffEntropy.dat');
%set F:
trainingDataANFIS = load('trainDataANFIS_contrast_IDM_diffVar.dat');
```

```
testingDataANFIS = load('testDataANFIS_contrast_IDM_diffVar.dat');

%OLD
%trainingDataANFIS = load('trainDataANFIS_variance_corr.dat');
%testingDataANFIS = load('testDataANFIS_variance_corr.dat');
%trainingDataANFIS = load('trainDataANFIS_mean_IDM_DE.dat');
%testingDataANFIS = load('testDataANFIS_mean_IDM_DE.dat');
%trainingDataANFIS = load('trainDataANFIS_contrast_sumV_imc12.dat');
%testingDataANFIS = load('testDataANFIS_contrast_sumV_imc12.dat');

%load trainDataANFIS_mean_sumAve.dat
%load testDataANFIS_mean_sumAve.dat

%fismat = genfis1(trainingDataANFIS); %default mf per input = 2 gbellmf
fismat = genfis1(trainingDataANFIS,3,'gbellmf'); %specify 3 gbellmf per input
numberOfInputs = size(trainingDataANFIS,2) - 1;
trnopt(1) = 50; % trnopt(1) = number of training epochs, default: 10
[fismat1,error1,ss,fismat2,error2] = anfis(trainingDataANFIS,fismat,trnopt(1),[],testingDataANFIS);
anfis_output1 = evalfis(trainingDataANFIS(:,1:end-1), fismat2);
anfis_output2 = evalfis(testingDataANFIS(:,1:end-1), fismat2);

%display in the command line the FIS rules
showrule(fismat2)

%display ANFIS diagram
figure, plotfis(fismat2)

%plot membership functions before training...individually
%figure
%for a = 1:numberOfInputs
%    subplot(numberOfInputs,1,a)
%    plotmf(fismat, 'input', a)
%end

%plot membership functions after training...individually
%figure
%for a = 1:numberOfInputs
%    subplot(numberOfInputs,1,a)
%    plotmf(fismat2, 'input', a)
%end

%PLOT MEMBERSHIP FUNCTIONS BEFORE TRAINING... IN ONE GROUP PLOT
figure
for a = 1:numberOfInputs
   subplot(numberOfInputs,2,a*2-1)
   plotmf(fismat, 'input', a)
   subplot(numberOfInputs,2,a*2)
   plotmf(fismat2, 'input', a)
end

%plot root mean squared error1 and error2 for training and testing data sets....
figure, plot(error1,'-k*');
hold on, plot(error2,'-k^');
legend('training data','testing data');
hold off
xlabel('training epochs');
ylabel('ANFIS root mean squared error');

%PLOT PREDICTED VALUES WITH EXPECTED VALUES FOR TRAINING DATA....
figure, subplot(1,2,1), plot(anfis_output1,'k+')
hold on, plot(trainingDataANFIS(:,end),'ko'), hold off
legend('predicted','expected');
xlabel('training data index');
ylabel('classification');
```

```
text(100,0,'classification:');
text(100,-0.1,'0.0 - normal, 0.5 - polyp, 1.0 - cancerous');

%PLOT PREDICTED VALUES WITH EXPECTED VALUES FOR TESTING DATA... AT THE RIGHT SIDE
%OF FIGURE ABOVE:
subplot(1,2,2), plot(anfis_output2,'k+')
hold on, plot(testingDataANFIS(:,end),'ko'), hold off
legend('predicted','expected');
xlabel('testing data index');
ylabel('classification');
text(3,1.22,'classification: 0.0 - normal, 0.5 - polyp, 1.0 - cancerous');


%PLOT (PREDICTED VALUE - EXPECTED VALUE) OF TRAINING DATA.....
figure, subplot(1,2,1), plot(anfis_output1-trainingDataANFIS(:,end),'-k.')
axis([0 210 -2 2])
xlabel('training data index')
ylabel('predicted value - expected value')
grid on
text(10,-1.25,'cancerous')
text(90,-1.25,'normal')
text(150,-1.25,'polyp')

%PLOT (PREDICTED VALUE - EXPECTED VALUE) OF TESTING DATA....AT THE RIGHT SIDE
subplot(1,2,2), plot(anfis_output2-testingDataANFIS(:,end),'-k.')
axis([0 90 -2 2])
xlabel('testing data index')
ylabel('predicted value - expected value')
grid on
text(8,-1.25,'cancerous')
text(38,-1.25,'normal')
text(68,-1.25,'polyp')


%produce the Mean Relative Differences Confusion Matrix (MRDCM) for the training data set....
trainingDataMRDCM = zeros(3);
for a = 1:210
   value1 = abs(anfis_output1(a) - 0.0);
   value2 = abs(anfis_output1(a) - 0.5);
   value3 = abs(anfis_output1(a) - 1.0);
   if trainingDataANFIS(a,end) == 0.0
      trainingDataMRDCM(:,1) = trainingDataMRDCM(:,1) + [value1; value2; value3];
   elseif trainingDataANFIS(a,end) == 0.5
      trainingDataMRDCM(:,2) = trainingDataMRDCM(:,2) + [value1; value2; value3];
   elseif trainingDataANFIS(a,end) == 1.0
      trainingDataMRDCM(:,3) = trainingDataMRDCM(:,3) + [value1; value2; value3];
   end
   clear value1 value2 value3
end
trainingDataMRDCM = trainingDataMRDCM./(size(trainingDataANFIS,1)/size(trainingDataANFIS,2))


%produce the Mean Relative Differences Confusion Matrix (MRDCM) for testing data set....
testingDataMRDCM = zeros(3);
for a = 1:90
   if testingDataANFIS(a,end) == 0.0
      value1 = abs(anfis_output2(a) - 0.0);
      value2 = abs(anfis_output2(a) - 0.5);
      value3 = abs(anfis_output2(a) - 1.0);
      testingDataMRDCM(:,1) = testingDataMRDCM(:,1) + [value1; value2; value3];
   elseif testingDataANFIS(a,end) == 0.5
      value1 = abs(anfis_output2(a) - 0.0);
      value2 = abs(anfis_output2(a) - 0.5);
      value3 = abs(anfis_output2(a) - 1.0);
```

```
        testingDataMRDCM(:,2) = testingDataMRDCM(:,2) + [value1; value2; value3];
    elseif testingDataANFIS(a,end) == 1.0
        value1 = abs(anfis_output2(a) - 0.0);
        value2 = abs(anfis_output2(a) - 0.5);
        value3 = abs(anfis_output2(a) - 1.0);
        testingDataMRDCM(:,3) = testingDataMRDCM(:,3) + [value1; value2; value3];
    end
    clear value1 value2 value3
end
testingDataMRDCM = testingDataMRDCM./(size(testingDataANFIS,1)/size(testingDataANFIS,2))


%produce the Confusion Matrix for the training data set:
threshold1 = 0.25; %threshold between normal and aden. polyp cases
threshold2 = 0.75; %threshold between aden. polyp and cancerous
trainingDataCM = zeros(3);
for a = 1:210
    if trainingDataANFIS(a,end) == 0.0
        if anfis_output1(a) < threshold1
            trainingDataCM(1,1) = trainingDataCM(1,1) + 1;
        elseif anfis_output1(a) < threshold2
            trainingDataCM(2,1) = trainingDataCM(2,1) + 1;
        else
            trainingDataCM(3,1) = trainingDataCM(3,1) + 1;
        end
    elseif trainingDataANFIS(a,end) == 0.5
        if anfis_output1(a) < threshold1
            trainingDataCM(1,2) = trainingDataCM(1,2) + 1;
        elseif anfis_output1(a) < threshold2
            trainingDataCM(2,2) = trainingDataCM(2,2) + 1;
        else
            trainingDataCM(3,2) = trainingDataCM(3,2) + 1;
        end
    elseif trainingDataANFIS(a,end) == 1.0
        if anfis_output1(a) < threshold1
            trainingDataCM(1,3) = trainingDataCM(1,3) + 1;
        elseif anfis_output1(a) < threshold2
            trainingDataCM(2,3) = trainingDataCM(2,3) + 1;
        else
            trainingDataCM(3,3) = trainingDataCM(3,3) + 1;
        end
    end
end
trainingDataCM

%produce the Confusion Matrix for the testing data set:
testingDataCM = zeros(3);
for a = 1:90
    if testingDataANFIS(a,end) == 0.0
        if anfis_output2(a) < threshold1
            testingDataCM(1,1) = testingDataCM(1,1) + 1;
        elseif anfis_output2(a) < threshold2
            testingDataCM(2,1) = testingDataCM(2,1) + 1;
        else
            testingDataCM(3,1) = testingDataCM(3,1) + 1;
        end
    elseif testingDataANFIS(a,end) == 0.5
        if anfis_output2(a) < threshold1
            testingDataCM(1,2) = testingDataCM(1,2) + 1;
        elseif anfis_output2(a) < threshold2
            testingDataCM(2,2) = testingDataCM(2,2) + 1;
        else
            testingDataCM(3,2) = testingDataCM(3,2) + 1;
        end
```

```
    elseif testingDataANFIS(a,end) == 1.0
        if anfis_output2(a) < threshold1
            testingDataCM(1,3) = testingDataCM(1,3) + 1;
        elseif anfis_output2(a) < threshold2
            testingDataCM(2,3) = testingDataCM(2,3) + 1;
        else
            testingDataCM(3,3) = testingDataCM(3,3) + 1;
        end
    end
end
testingDataCM

%FACTOR MATRIX
fm = [1/3 -0.3 -0.5;
    -0.05 1/3 -0.4;
    -0.2 -0.1 1/3];

%NORMALISED CONFUSION MATRICES
nTrainingDataCM = trainingDataCM./repmat(sum(trainingDataCM),3,1);
nTestingDataCM = testingDataCM./repmat(sum(testingDataCM),3,1);

%CLASSIFICATION PERFORMANCE INDEX
trainingDataCPI =  sum(sum(fm.*nTrainingDataCM))
testingDataCPI =  sum(sum(fm.*nTestingDataCM))

%CLASSIFICATION PERCENT ACCURACY
trainingDataPA = trace(nTrainingDataCM)/sum(nTrainingDataCM(:))*100
testingDataPA = trace(nTestingDataCM)/sum(nTestingDataCM(:))*100

%SAVE ALL VARIABLES USED....
%save('C:\Documents and Settings\lagrange\My Documents\My Dropbox\dissertation\ANFIS run saved MATLAB variables\variablesANFIS2.mat');
save(strcat(fileSaveLocation,fileSaveName));

fprintf('\ntype "showrule(fismat2)" in the command line to show the rules\n')
```

A.8 Image Classification Test for Pathologists:

Below are print-outs of the images used in the survey test for pathologists. In the actual survey/test form, each A4-sized page contained 3 images. Each image was printed to have dimensions of approximately $3\,{}^{1}/_{8}$ inches by $4\,{}^{3}/_{16}$ inches.

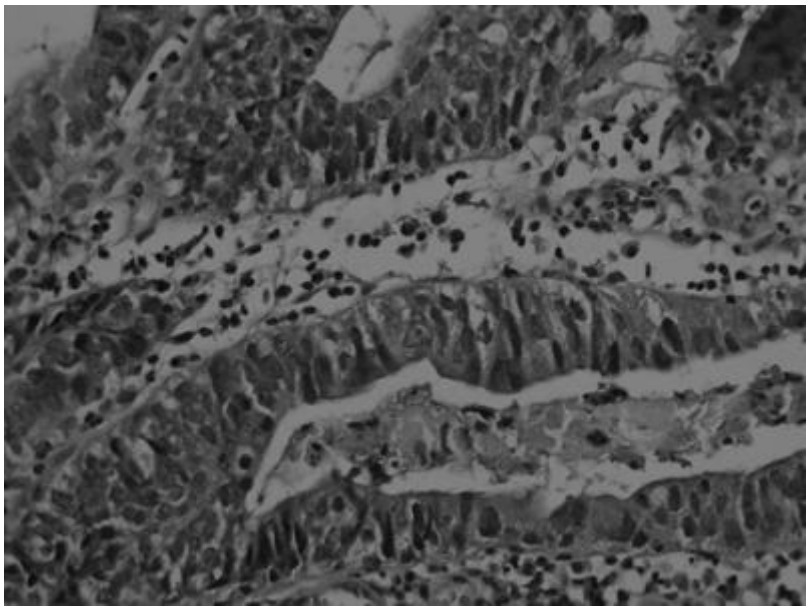


Image 1:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

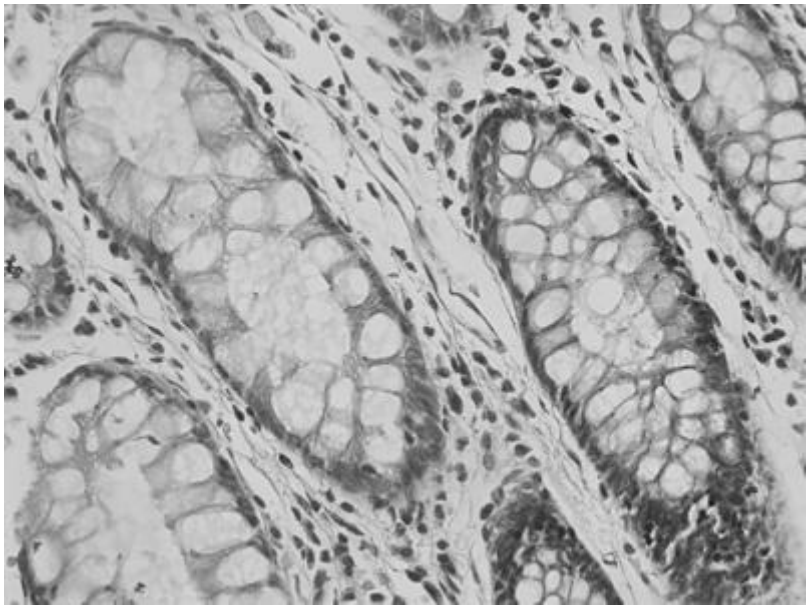


Image 2:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

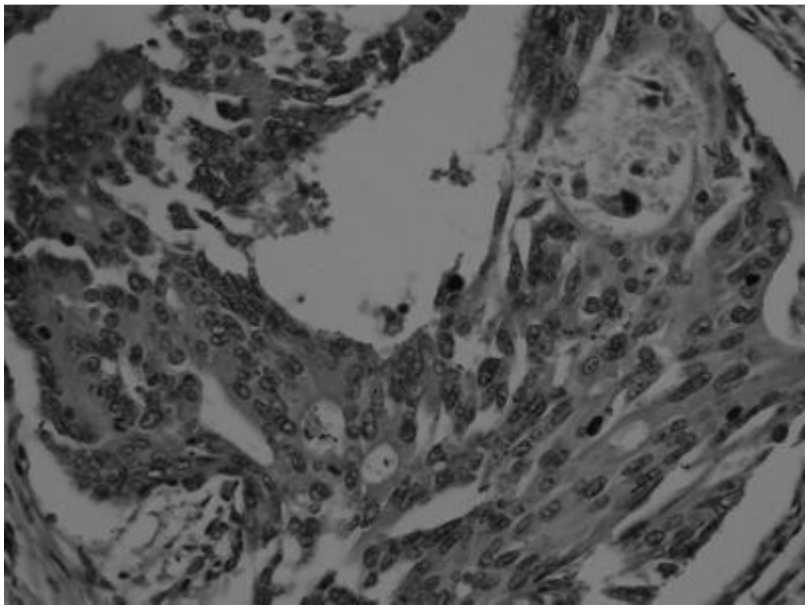


Image 3:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

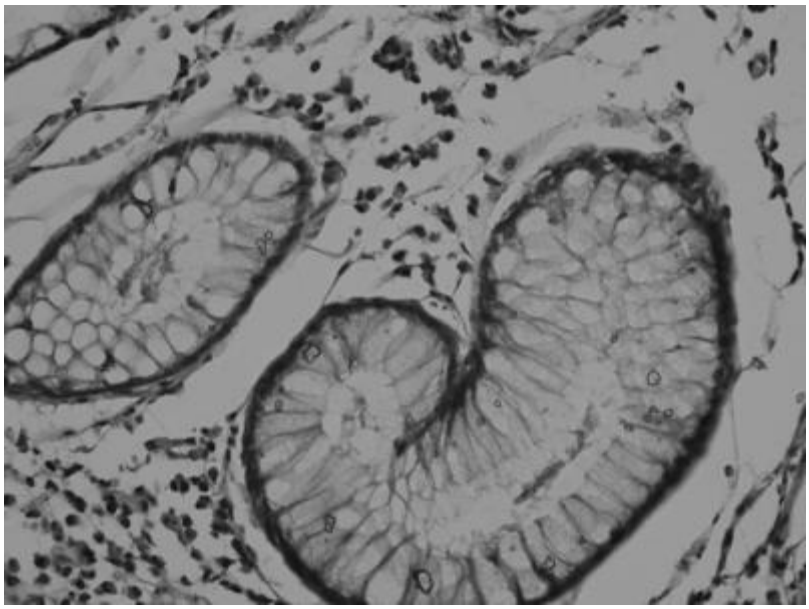


Image 4:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

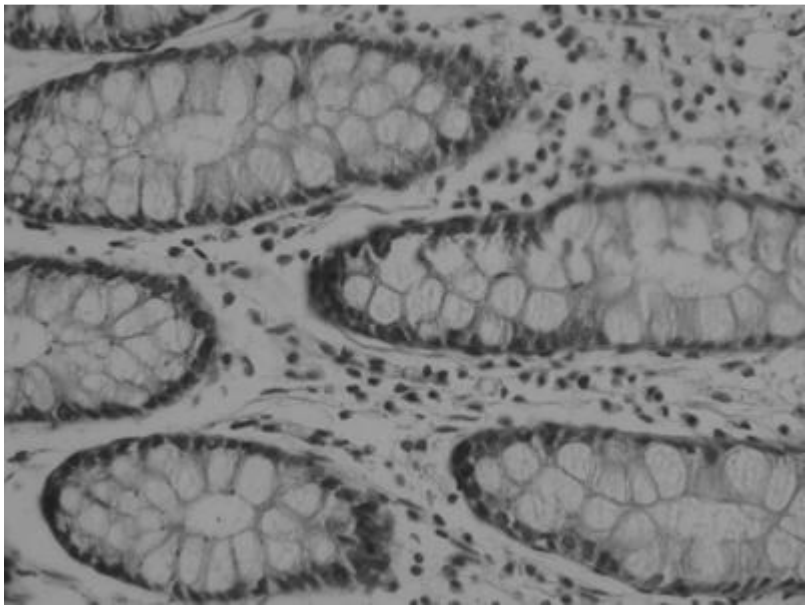


Image 5:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

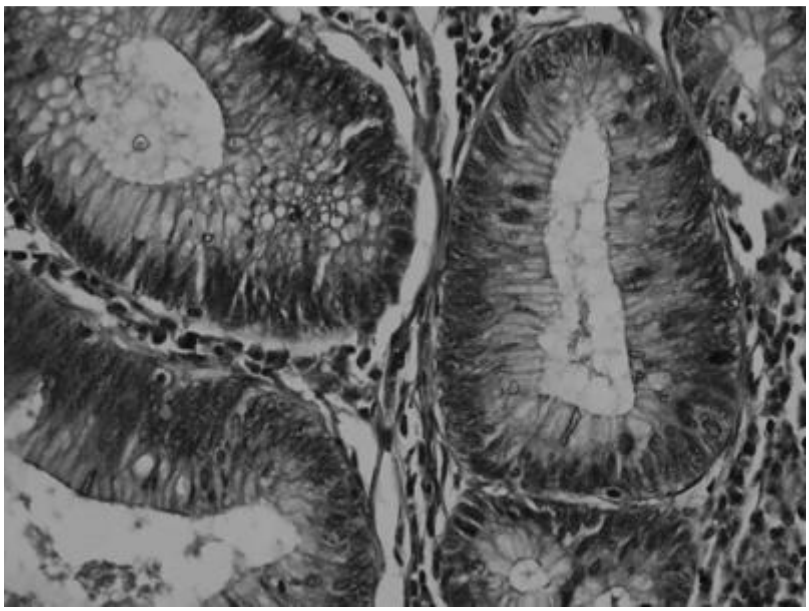


Image 6:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

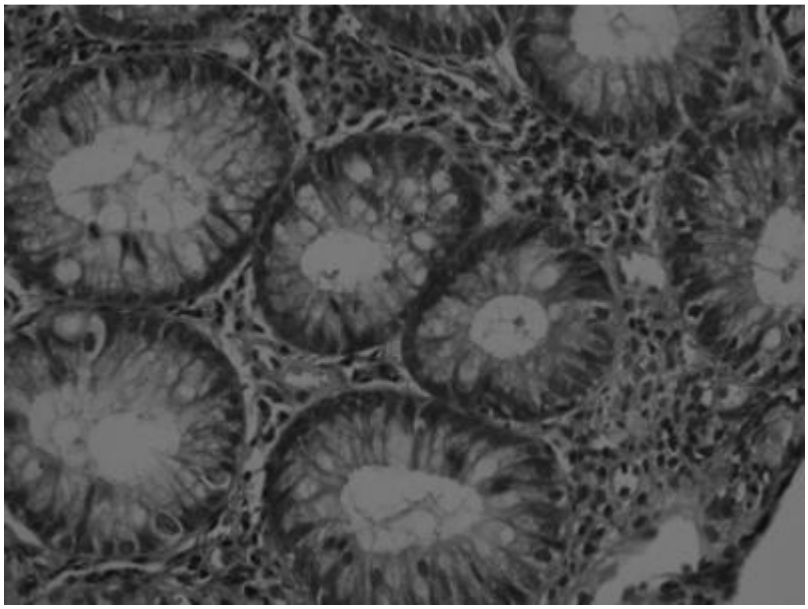


Image 7:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

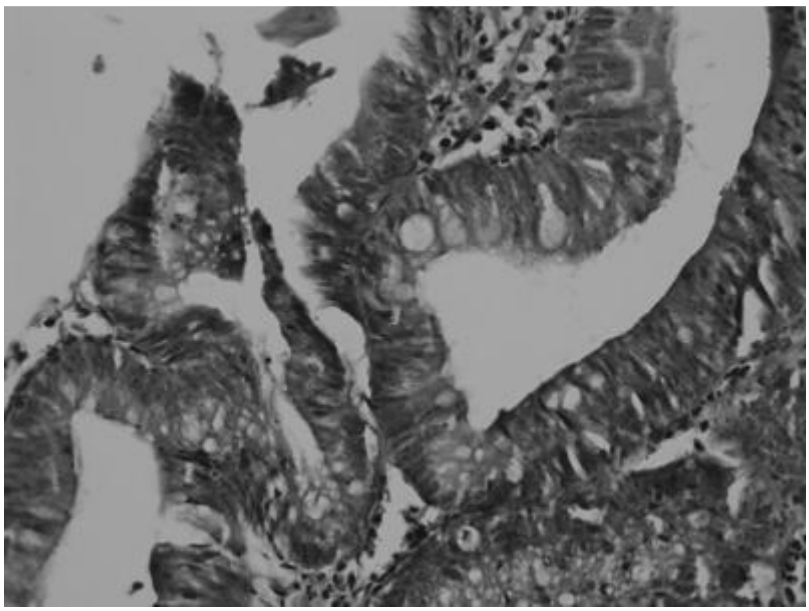


Image 8:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

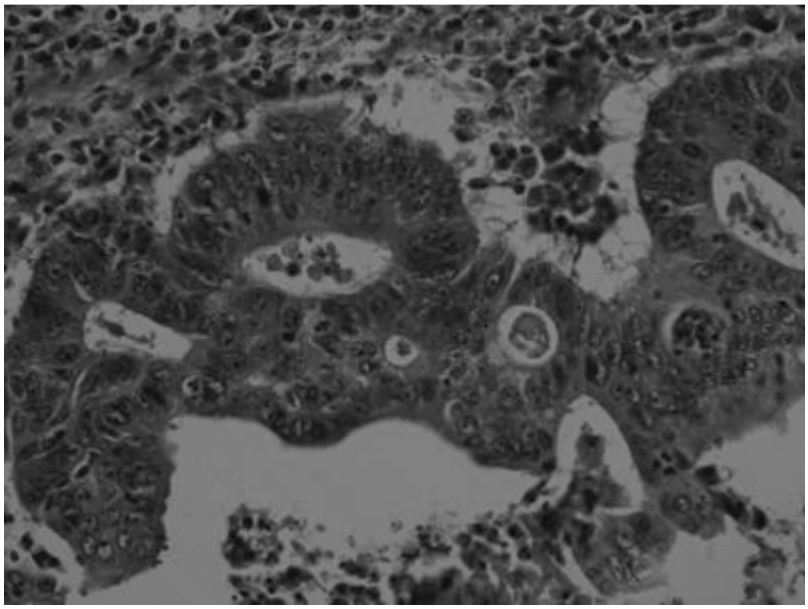


Image 9:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

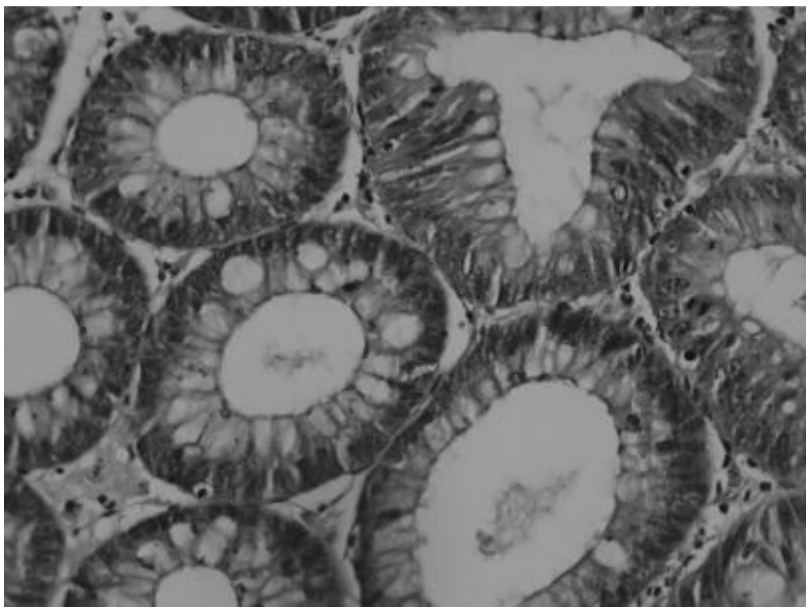


Image 10:

- ⬭ Normal
- ⬭ Aden. Polyp
- ⬭ Cancerous

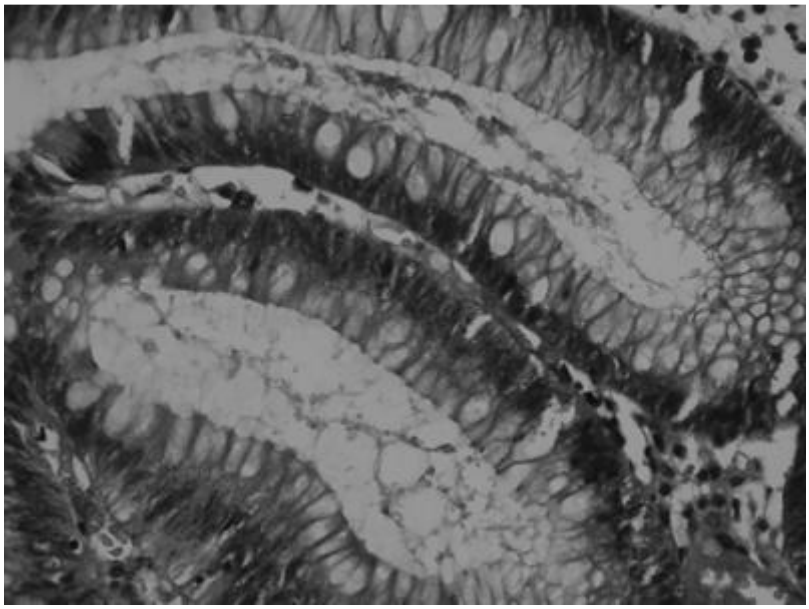


Image 11:

○ Normal

○ Aden. Polyp

○ Cancerous

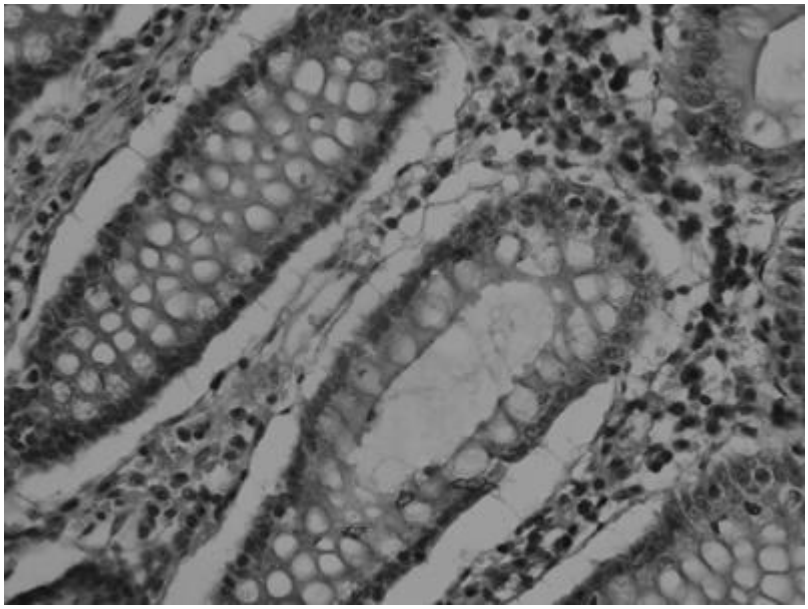


Image 12:

○ Normal

○ Aden. Polyp

○ Cancerous

Image 13:

⬭ Normal

⬭ Aden. Polyp

⬭ Cancerous

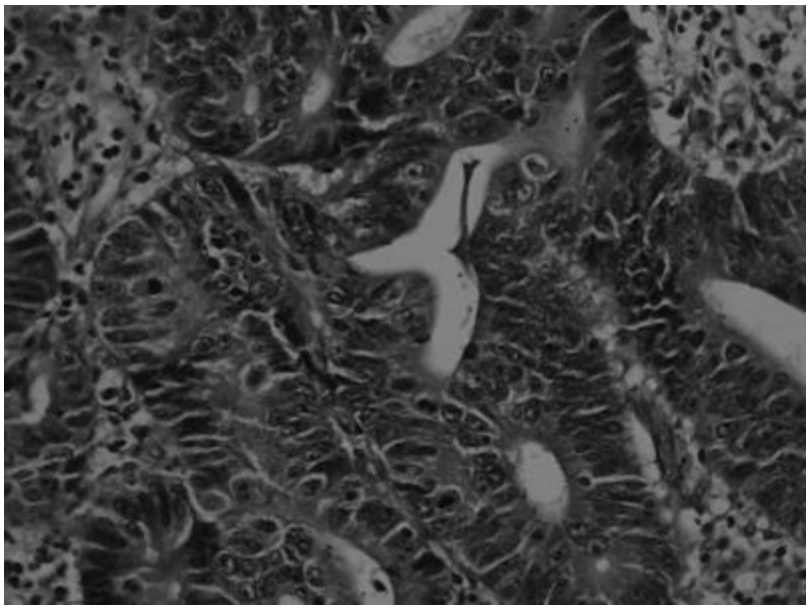


Image 14:

⬭ Normal

⬭ Aden. Polyp

⬭ Cancerous

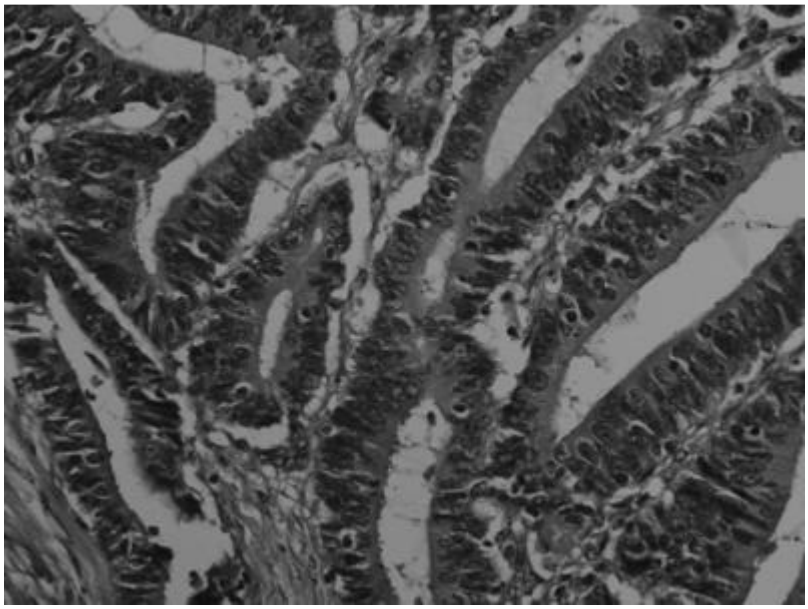


Image 15:

⬭ Normal

⬭ Aden. Polyp

⬭ Cancerous

A.9 Expected classes of the images used in the classification test for pathologists:

The table below outlines the 'true' classes of the images that were used in the performance test on the pathologists.

Image number	Expected Class
1	Cancerous
2	Normal
3	Cancerous
4	Normal
5	Normal
6	Adenomatous Polyp
7	Adenomatous Polyp
8	Adenomatous Polyp
9	Cancerous
10	Adenomatous Polyp
11	Adenomatous Polyp
12	Normal
13	Normal
14	Cancerous
15	Cancerous

A.10 Summary of Publications Produced Based on this Research

1. Gan Lim, L A, Naguib, R N G, Dadios, E P, dela Fuente, D (2007). 'Methodology for the Development of an Automatic Classifier for Colonic Mucosa Microscopic Images Using Hybrid GA-NN-FL'. Proc. of the 3rd International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM). Century Park Hotel, Manila, Philippines. March 15-18, 2007.

2. Gan Lim, L A, Naguib, R N G, Dadios, E P, dela Fuente, D (2007). 'Useful GLCM Textural Properties in the Classification of Colonic Mucosa Microscopic Images'. 4th Pacific Asia Conference on Mechanical Engineering (PACME). Bayview Hotel, Roxas Blvd., Manila. August 28-29, 2007.

3. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2008). 'Using K-Means Clustering to Classify Microscopic Colon Images'. Proc. of the 10th Science and Technology Congress 2008, DLSU, Manila, July 23, 2008.

4. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2008). 'Identification of Cancerous Microscopic Colonic Images Using Neural Networks'. Proc. of the 11th Osaka University - DLSU Academic Research Symposium, De La Salle University, Manila, Sept. 1-5, 2008.

5. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2009). 'Using Hyperchromasia to Classify Colonic Microscopic Images'. Proc. of the 5th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management

(HNICEM). Century Park Hotel, Manila, Philippines. Traders Hotel, Manila. March 12-15, 2009.

6. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2009). 'Classification of Colonic Histopathological Images Using Hough Transform'. Proc. of the Science and Technology Congress 2009, DLSU, Manila, Sept. 21-23, 2009.

7. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2009). 'Detection of Glands in Colonic Histopathological Images Using GLCM Texture'. Proc. of the National Electrical, Electronics, and Computing Conference (NEECC 2009), Science Discovery Center, SM Mall of Asia, December 9-11, 2009.

8. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2010). 'Image Classification of Microscopic Colonic Images Using Textural Properties and KSOM'. International Journal of Biomedical Engineering and Technology (IJBET), Vol. 3, Issue 3/4, pp. 308-318.

9. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2010). 'Analysis of Colonic Histopathological Images Using Pixel Intensities and Hough Transform'. Philippine Science Letters, Vol. 3, No. 1. (***2011 NAST Outstanding Paper Award***)

10. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2010). 'Using Genetic Algorithm and Artificial Neural Network to Classify Colonic Histopathological Images'. Proc. of the 15th Joint Academic Research Symposium of De La Salle University and Osaka University, Sept 29-30, 2010.

11. Gan Lim, L A, Naguib, R N G, Dadios, E P, Avila, J M C (2011). 'Using ANFIS to Classify Colonic Histopathological Images'. Book of Abstracts: Annual PAASE Meeting and Symposium (APAMS) 2011. National Institute of Physics, University of the Philippines, Manila. June 15-18, 2011