HEP-2 CELL IMAGES CLASSIFICATION BASED ON STATISTICAL TEXTURE ANALYSIS AND FUZZY LOGIC

by

NUR FARAHIM BINTI JAMIL

FINAL PROJECT REPORT

Submitted to the Department of Electrical and Electronic Engineering
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical and Electronic Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

HEP-2 CELL IMAGES CLASSIFICATION BASED ON STATISTICAL TEXTURE ANALYSIS AND FUZZY LOGIC

by

Nur Farahim binti Jamil

13315

A project dissertation submitted to the
Department of Electrical and Electronic Engineering
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical and Electronic Engineering)

Approved:

_____

Dr Ibrahima Faye
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

January, 2014

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

_____

Nur Farahim binti Jamil

ABSTRACT

Autoimmune diseases occur when an inappropriate immune response takes place and produces autoantibodies to fight against human antigens. In order to detect autoimmune disease, a test called indirect immunofluorescence (IIF) will be carried out to identify antinuclear autoantibodies (ANA) in the HEp-2 cell. The outcome of the test includes observing fluorescence intensity of the sample and classifying the staining pattern of the cell. Current method of analysing the results is limited to subjective factors such as experience and skill of the medical experts. The results obtained from the visual analysis are debatable as it is inconsistent. Thus, there is a need for an automated recognition system to reduce the variability and increase the reliability of the test results. Automated system also saves time and cost as the system is able to process large amount of image data at one time. This project proposes a pattern recognition algorithm consisting of statistical methods to extract seven textural features from the HEp-2 cell images followed by classification of staining patterns by using fuzzy logic. This method is applied to the data set of the ICPR 2012 contest in which each cell has been manually segmented and annotated by specialists. The textural features extracted are based on the first-order statistics and second-order statistics computed from grey level co-occurrence matrices (GLCM). The first-order statistics features are mean, standard deviation and entropy while the features extracted by GLCM are contrast, correlation, energy and homogeneity. The extracted features will then be used as an input parameter to classify the staining pattern of the HEp-2 cell images by using Fuzzy Logic. The staining patterns are divided into five categories; homogeneous, nucleolar, centromere, fine speckled and coarse speckled. A working classification algorithm is developed by using MATLAB and the Fuzzy Logic Toolbox to differentiate and classify the staining pattern of HEp-2 cell images. The algorithm gives a mean accuracy of 84% out of 125 test images.

ACKNOWLEDGEMENT

First and foremost, the author would like to express her utmost gratitude to God for His kind blessings by giving her the strength and determination to complete her Final Year Project. The author would also like to take this opportunity to express her gratitude to certain parties, organizations and individuals as this project may not be successful without their involvement and contributions. A big thank you to Dr. Ibrahima Faye for his advice, supervision and support throughout the entire Final Year Project execution. He has constantly provided guidance and valuable knowledge in each and every stage; from the planning phase of the project until the completion of this dissertation.

Special thanks as well to the co-supervisor, Ms. Zazilah May for her assistance in making this project a success. The author would also like to thank the FYP coordinators for conducting seminars and briefings that help FYP students in planning for the project development as well as their effort in arranging evaluation such as ElectrEx and Viva by inviting internal and external examiners from related field to evaluate the project. Apart from the parties and individuals above, the author would like to extend her personal gratitude to her colleagues and beloved family for their support and continuous encouragement throughout the duration of the project.

Last but not least, the author would like to thank all parties involved once again for making this Final Year Project a success.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF EQUATIONS

LIST OF FIGURES

LIST OF ABBREVIATIONS

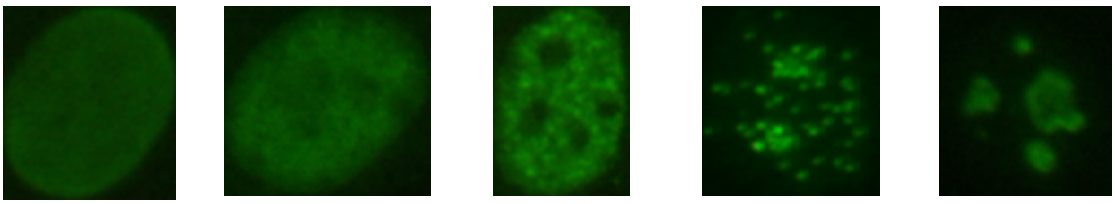
HEp-2	Human epithelial type 2
IIF	Indirect immunofluorescense
ANA	Antinuclear autoantibody
GLCM	Grey Level Co-occurrence Matrices
CAD	Computer Aided Diagnostic
ICPR	International Conference of Pattern Recognition
SVM	Support Vector Machine
CCD	Charge-coupled device
GUI	Graphical User Interface

CHAPTER 1
INTRODUCTION

1.1 Background Study

Millions of people all over the world are diagnosed with autoimmune disease, either from one or more than one from a list of 60 known diseases. Autoimmune diseases arise from inappropriate autoimmune response of the human body where the system malfunctions and produces antibodies to fight against human antigens. It is considered as a fatal disease as it may results in serious damage of tissue and organs. Examples of autoimmune disease include diabetes mellitus type I, multiple sclerosis and rheumatoid arthritis [1].

To diagnose autoimmune disease, it is crucial to identify ANA in HEp-2 cells. ANA is commonly detected by a test IIF. The fundamental part of the test is to observe a sample in a slide with HEp-2 cell line under a fluorescence microscope. Generally, the diagnostic procedure of an IIF test can be simplified into two steps. The first step involves observation of the fluorescence intensity followed by the classification of staining patterns [2]. With regards to fluorescence intensity classification, medical experts categorize the intensity into three main levels – positive, intermediate or negative. Classification of staining patterns ensues if the fluorescence intensity falls in either the intermediate or positive class [3]. Different staining patterns determine different type of autoimmune disease. Figure 1 [4] shows relevant patterns related to the most frequent ANAs while Table 1 describes the characteristics of each pattern [5].



(a) (b) (c) (d) (e)

Figure 1: Staining patterns (a) homogeneous (b) fine speckled (c) coarse speckled (d) centromere (e) nucleolar

Table 1: Characteristics of staining pattern

Pattern	Description
Homogeneous	Diffuse staining, consistent across the whole nucleus
Fine speckled	Fine granular nuclear staining
Coarse speckled	Coarse granular nuclear staining, larger specks
Centromere	Large numbers of strong bright spots on a darker ground
Nucleolar	Large speckled staining with less than six in number within the nucleus

In this particular project, the main focus is on the textural features extraction of the HEp-2 cell images and classifying the images into five staining pattern. Texture refers to the spatial distribution of grey levels and can be defined as the deterministic or random repetition of one or several basic patterns in an image [6]. Textural features can be represented in three ways; statistical, syntactic or structural and spectral [7]. Statistical approach is chosen as it is particularly useful when the basic patterns are small, providing better results for micro textures [8]. The classifier chosen for this project is the Fuzzy Logic technique which provides a simple way to conclude indistinct information as this technique is able to identify truth values that ranges between 0 and 1 [9]. In other words, it is able to process data that is approximate rather than fixed and exact.

1.2 Problem Statement

The concept of this project is to design and simulate a classifier for the staining pattern. Current techniques of pattern classification available in literature have some limitation in terms of performance and accuracy. The major disadvantages of the current IIF procedure can be summarized as follows:

a) Low level of standardization in IIF methodological procedures such as unsuitable microscope magnification for reading slides or non-conventional cutoff dilution of serum [10].

b) Test results depend on subjective factors such as medical knowledge or years of experience which results in high inter and intra laboratory inconsistency [11].
c) Photobleaching effect that bleaches the cells drastically over a short period of time [12].

The lack in automation leads to low performance as well as accuracy. Besides, visual analysis of large amount of images can be a monotonous and lengthy process. It is not beneficial for patients since the time consuming diagnostic procedure may delay the physicians' ability to perform any further medical actions. The manual visual analysis performed by experts is also at times dubious as the considered patterns are highly correlated with the degree of visual similarity to other patterns. Nucleolar and centromere patterns have similar visual appearance and are usually confused with each other. The same observation can be concluded for the other three patterns which are homogeneous, coarse speckled and fine speckled.

1.3 Significance of Project

This project mainly aims to develop an algorithm which is able to produce outputs that are consistent, sensitive and accurate in classifying the five staining patterns. This provides a solution for the inconsistency in the test results and over reliance on medical experts in performing manual visual analysis.

Besides, the development of an automated classification system reduces the time taken for tedious manual analysis as it able to analyse large volume of image data faster. This is favorable for patients to get the right medical attention. An automated system also increases the efficiency and effectiveness of the medical sector.

1.4 Objectives

There are three main objectives that need to be achieved at the end of this project. The objectives are:

1. To identify textural features by statistical approaches to be utilized in pattern classification using Fuzzy Logic.
2. To develop classification algorithm using identified textural features and Fuzzy Logic.
3. To assess the performance of classification algorithm and to validate the accuracy of the classification results.

Thus, this research aims to develop an automated classification system that is able to identify features correctly and in turn results in accurate pattern classification.

1.5 Scope of Study

To achieve the objectives and ensure the feasibility of this project, it is vital to narrow down the scope of study from the extensive research that has been conducted. The scope of study is divided into three parts - staining pattern, statistical methods to extract textural features and Fuzzy Logic. From there, the main subjects under investigation are:

i. Understanding and learning MATLAB application in image processing particularly on feature extraction and pattern recognition
ii. Identifying statistical approaches that are most suitable for the project
iii. Implementing fuzzy logic as a classifier and familiarize with Fuzzy Logic Toolbox in MATLAB

1.5.1 Staining Pattern

With regards to the fluorescence staining pattern of the HEp-2 cell images, there are more than 30 different known fluorescence patterns which are categorized into a set of 100 different autoantibodies [1]. To classify 30 different pattern would be a time-

consuming operation. Hence, for this project, only five of the nucleus staining patterns are considered for classification. The patterns are homogeneous, nucleolar, centromere, fine speckled and coarse speckled. These patterns are chosen as they are characterized by well-defined edges.

1.5.2 Statistical Approach to Derive Textural Features

Texture depends on its tone and structure. Based on this, it may be fine, coarse, smooth or grained. Tone is the properties of pixel intensity while structure is the spatial relationship between basic patterns [13]. To extract the textural features, two statistical methods are selected; first-order statistics [7] and second-orders statistics computed from grey level co-occurrence matrices (GLCM) [14-16].

From the first-order statistics, three textural features will be extracted which are mean, standard deviation and entropy. Mean is a measure of brightness, standard deviation measures contrast while entropy is a measure of randomness in which large entropy signifies that the image is not texturally uniform.

GLCM is a frequently used statistical method to analyse texture. It extracts texture information of an image from the spatial dependence of grey level values. Specifically, the perception of textures are computed from a set of grey-tone spatial dependence matrices by how often a pixel with a grey level value occurs horizontally adjacent to its neighbouring pixel according to different angles and distance [2]. Four features will be extracted from second-order statistics computed from GLCM. The features are contrast, angular second moment, correlation and homogeneity. Contrast provides a measure on the sharpness of structural variations in an image or the difference in intensity between neighbouring regions. High contrast image would show visible and separable areas of different intensity. Angular second moment or energy measures the smoothness of an image by uniformity of the pixel pair repetitions. Correlation depicts the grey linear dependency to show the linear relationship between a pixel to its neighbour over the whole image. Homogeneity is the opposite of contrast which implies that homogeneity expands with less contrast. There are seven textural

features to be extracted in total and these features will then be used as an input parameter to the classifier.

1.5.3 Fuzzy Logic

Based on [17], fuzzy logic has a few advantages. It is tolerant to inexact data, able to model nonlinear functions of arbitrary complexity, flexible, easily understood and is built on natural language. In this case where it is required to map an input parameters to output parameters, fuzzy logic provides a convenient solution.

There are two types of fuzzy logic inference system that can be used: Mamdani-type and Sugeno-type [17]. Mamdani is the method which is frequently used. The initial two stages of the process which are fuzzifying the inputs and applying the fuzzy operator are similar for both types. The difference lies in how the output is determined. Mamdani expects the output membership functions to be fuzzy sets and requires defuzzification. Sugeno, on the other hand, models any system in which the output membership functions are either linear or constant.

Figure 2 shows the Mamdani fuzzy inference system which consists of a fuzzification interface, a rule base, a database, a decision-making unit and a defuzzification interface.

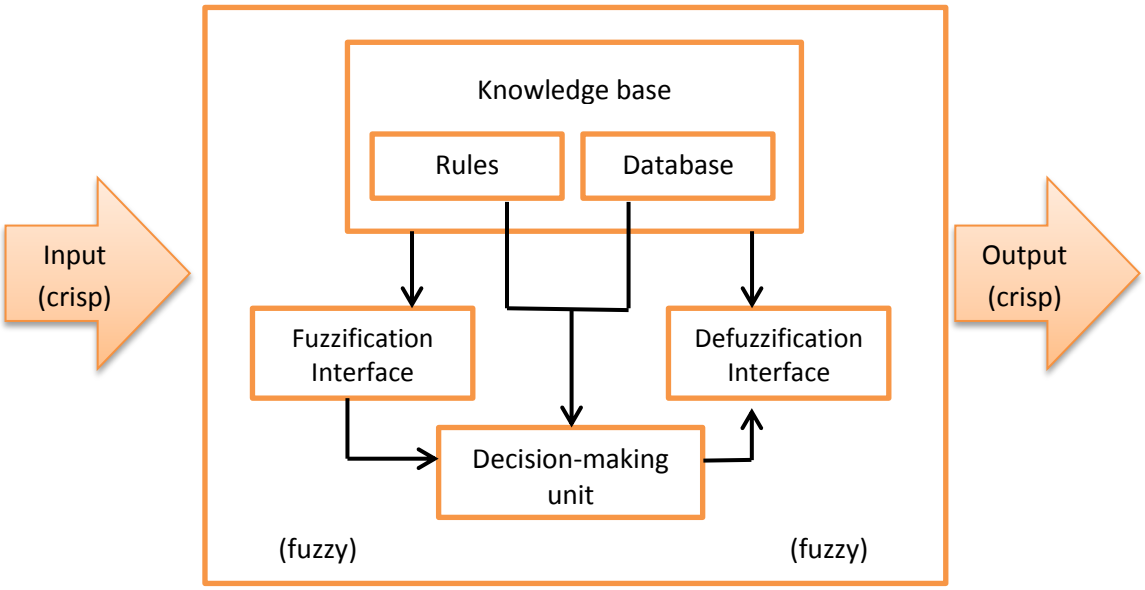


Figure 2: Mamdani fuzzy inference system

The fuzzification interface transforms the crisp inputs into degrees of match with linguistic values, the rule base contains the number of fuzzy IF-THEN rules, the database defines the membership function of the fuzzy sets used in the fuzzy rules, the decision-making unit performs the inference operation on the rules and the defuzzification interface transforms the fuzzy results of the inference into a crisp output.

Figure 3 shows the rule-based structure of fuzzy logic to implement the if X AND/OR Y then Z rules in which the desired system output response is defined by the system input conditions [9]. The input conditions, in this case, would be the textural features and the output response would be the five staining patterns.

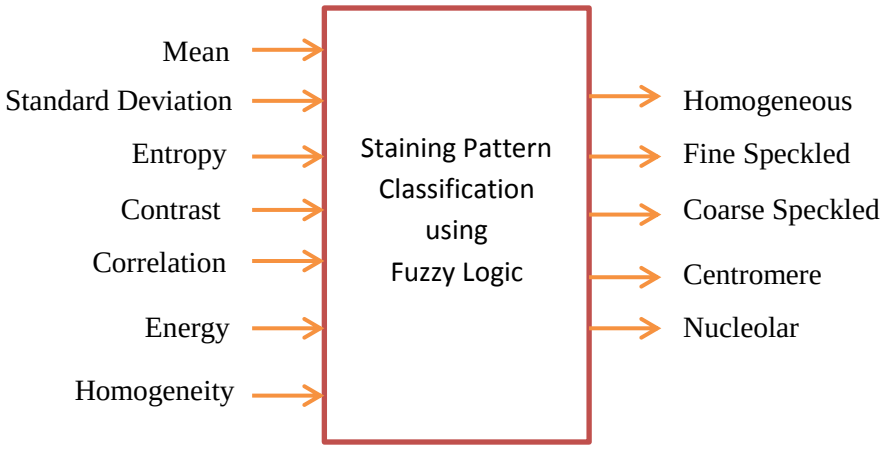


Figure 3: Rule-based structure

1.6 Relevancy of Project

There are many applications that have been developed to classify HEp-2 cell images but none of the available literature uses a combination of first-order statistics and GLCM as descriptors as well as Fuzzy Logic as a mean of classifier. Consequently, this project is relevant. It deals mainly with image processing which requires technical knowledge in programming and is very much related to four years undergraduate study majoring in electrical and electronics engineering. Project management skills such as time management and interpersonal communication skill are needed as well. This project also challenges critical, analytical and innovative thinking. All of which are required in the real working environment of a professional, competent and qualified engineer.

1.7 Feasibility of Project

The feasibility of this project is evaluated based on the technical aspects, economical values and project duration. Technical feasibility considers the process of learning, assimilating and applying the knowledge learnt. Basically, this project is feasible technically as it is simulation based which requires basic programming knowledge. Besides, before attempting to simulate in MATLAB, the author is required to conduct extensive research to identify the most practical approaches and techniques in order to be able to classify the patterns. In terms of project budget, it can be concluded that this project does not require any expenses as the software is readily available in the laboratory and the image dataset can be downloaded online. The project is completed within 28 weeks in which the first 14 weeks focuses more on the project planning, literature review and preliminary results while the remaining 14 weeks of the project period concentrates more on the continuation of the preliminary results to develop the classification algorithm. Hence, the project is feasible from the technical, time, and economical aspects.

CHAPTER 2

LITERATURE REVIEW

2.1 Statistical Method

There are various statistical methods to extract textural features. Table 2 summarizes possible statistical method to extract textural features.

Table 2: Statistical method to extract textural features

Author	Technique	Applications	Advantages	Disadvantages
G. Srinivasan and G. Shobha [18]	First order statistics	Obtained from original image values	Easy to analyse characteristics of image from histogram	Does not provide information on relative position of pixels to each other
	Spatial frequencies	Define texture by the number and types of primitives	Able to characterize texture as fine or coarse based on tone and structure characteristics of the primitives	Same number, type of basic patterns and spatial relationship does not imply same texture
	Law's texture energy measures	Determine texture properties by assessing average grey level, edges, spots, ripples and waves	Derive 5 texture properties from three simple vectors	Long computation
A. Materka and M. Strzelecki [19]	GLCM	Estimate image properties by taking into account relationship between pixels or group of pixels	• Consider spatial properties • Simple	Does not consider primitive shape
	Fractal-based	Provides a description on the complexity or irregularity of an object	High relation with human judgement to surface roughness	Uniform in all orientations
R. Maini and H. Aggarwal [20]	Edge frequency	Measures the total length of all edges in a region to measure coarseness or complexity of a texture	Usage of simple operators such as Robert's and Sobel's to analyse textures	Depends on the size of object and sensitive to noise

From the existing methods, two of the methods are selected for this project. The firs-order statistics estimate properties while ignoring the spatial interaction between image pixels. The second-order statistics estimate properties of two pixels values occurring at specific locations relative to each other which means that it takes into consideration the different angles and distances of neighbouring pixels. The combination of these two methods provides high discriminative power to distinguish different kind of images.

2.2 Existing Computer Aided Diagnostic (CAD) Systems

There has been an increasing interest in utilizing image processing technique for various clinical tests in recent years. As mentioned before, IIF test results depend on subjective factors resulting in low reproducibility and high variability. To overcome this problem, CAD is implemented to automatically determine the HEp-2 cells staining pattern. There was a HEp-2 cells classification contest held at the 2012 edition of the International Conference on Pattern Recognition (ICPR) in Tsukuba, Japan that focuses on IIF image analysis [21]. From the contest, 28 different recognition systems have been proposed and are able to automatically classify the staining pattern. Out of the 28 entries, three of the proposed methods achieve the highest performance. Table 3 summarizes the 3 classification method.

Table 3: Classification Method

Author	Classifier	Techniques	Advantages	Disadvantages	Results
K. Li, J. Yin et al. [22]	Multiclass boosting support vector machine (SVM)	• Uses 4 texture descriptors and utilizes multiclass probability SVM for the four feature sets. • Merge and integrate 4 SVMs as classifier with AdaBoost.M1 embed in the algorithm	Gives probability vectors as output rather than hard labels in comparison to traditional SVM	Unable to classify weak patterns	77% accuracy on image level
R. Nosaka, Y. Ohkawa, and K. Fukui [23]	Linear SVM classifier	• Uses only the green channel • Filtered using Gaussian function to suppress noise • Uses CoALBP to describe complex textures	Robust against disparity in illuminations and includes spatial relationship	Misclassify 2 images out of 14 images	85% accuracy on image level
X. Kong, K. Li et al.[24]	k-nN with χ^2 distance	• Extract statistical intensity features • Filter using k-means clustering • Represent each image by frequency histogram	Classify single images using few models without information about the image conditions	Unable to classify weak patterns	77% accuracy on image level

2.3 Existing Fuzzy Logic Application

Fuzzy Logic system has been implemented in several applications specifically addressing the classification problem in various fields. This is due to its ability to solve a complex nonlinear problem by expressing it in a linguistic form [25]. Table 4 shows application of fuzzy logic as a classifier.

Table 4: Fuzzy Logic Classifier

Author	Applications	Techniques	Advantages	Disadvantages	Results
G. Schaefer, T. Nakashima et al.[26]	Diagnosis of breast cancer	• Cancer is detected using thermography • Fuzzy rule-based classification system for diagnosis is built by using the statistical features extracted by the thermogram	Classification accuracy increases as the number of fuzzy partitions used increases	Does not achieve perfect classification even though classifiers are tested on the same data that was used on training	Classification rate falls just below 80% which is comparable to other techniques for breast cancer diagnosis such as mammography and ultrasonography
E. Sivasankar and R. Rajesh [9]	Severity of appendicitis in patients with right iliac fossa (RIF) pain	• Assessed by a Fuzzy Logic rule based classifier which consists of a fuzzification interface, a rule base, a database, a decision making unit and a defuzzification interface • A total of 11 input parameters which leads to 3 output parameters consisting of mild, moderate and severe appendicitis	Designed for cases where outputs are vague	Dimensionality as ratio of features to sample is large	Accuracy rate of • 96.55% for patients suffering with mild appendicitis • 94.71% for patients suffering with moderate appendicitis • 94.11% for patients suffering with severe appendicitis

CHAPTER 3

METHODOLOGY

3.1 Project activities

Figure 4 illustrates the overview of project activities flow for Final Year Project I (FYPI) as highlighted in purple and the continuation of project flow for Final Year Project II as seen in blue.

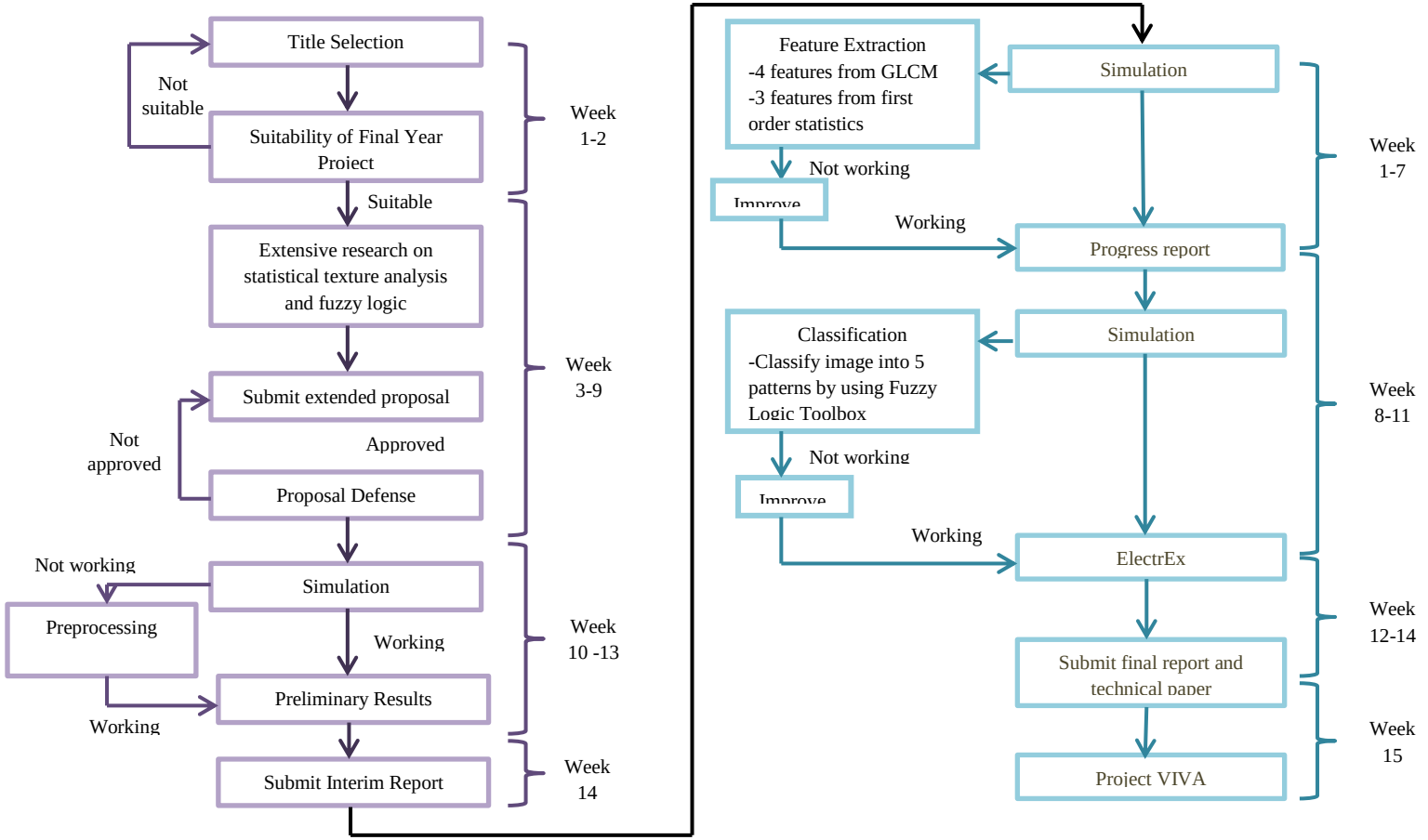


Figure 4: Project Activities flow chart

3.2 Research Methodology

3.2.1 Image Processing Flow

With regards to processing the HEp-2 cell images, there is a flow which is a fundamental analysis chain of image processing by an automated system. Figure 5 illustrates the general image processing flow.

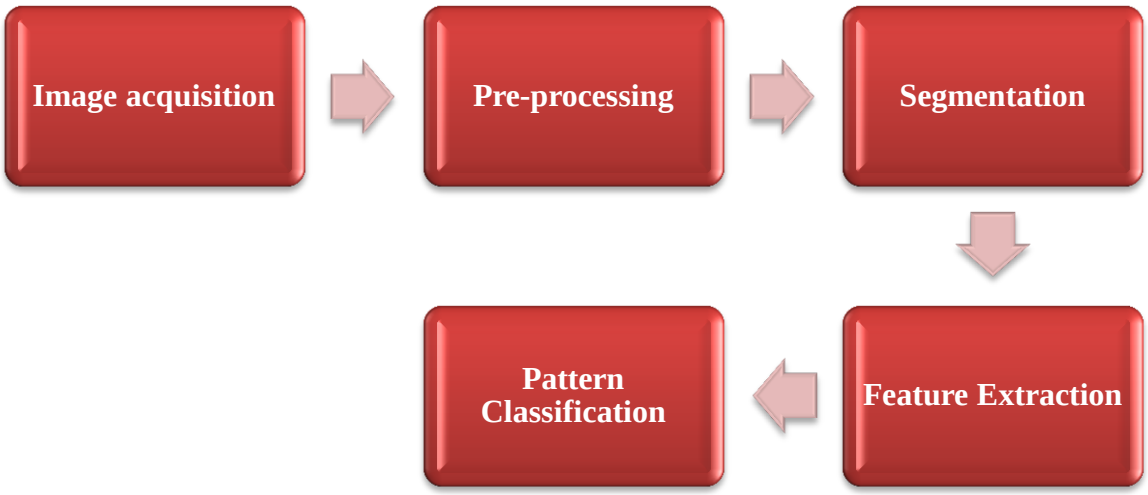


Figure 5: Image processing flow

3.2.2 Image Acquisition

The image is attained from the MIVIA HEp-2 images dataset which is an outcome of a research project [4]. The dataset contains images obtained from slides of Hep-2 substrate at the fixed dilution of 1:80. The images were acquired using a fluorescence microscope (40-fold magnification) coupled with a 50W mercury vapor lamp and with a digital camera. The camera has a charge-coupled device (CCD) with squared pixel of equal side to 6.45 µm. The images have a resolution of 1388×1038 pixels, a color depth of 24 bits and are stored as bitmap images. The dataset also has information on the staining pattern and intensity of the whole image.

3.2.3 Pre-processing

The purpose of image pre-processing is to reduce any unwanted noise and to improve image features that are necessary for further analysis. However, in the analysis of medical images, image pre-processing is not recommended as it may decrease the image information content [7]. Thus, in this case, the only pre-processing step that is carried out is converting the HEp-2 images to grey-scale.

3.2.4 Segmentation

Segmentation is an important process of isolating an image into regions with similar properties such as grey level and texture. The images obtained from the MIVIA dataset have been manually segmented by the medical specialists.

3.2.5 Feature Extraction

Once the image has been pre-processed and segmented, the textural features will then be extracted by using the first-order and second-order statistics computed from GLCM to characterize the staining pattern. A total of seven features will be extracted which will be the input parameters for the classification stage.

The first feature, mean is calculated in MATLAB by taking the mean values of the elements along different dimensions of an array. The standard deviation is obtained using the formula in Equation 1 where x is the data vector and n is the number of elements in the sample.

$$Standard\ Deviation = \left(\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \bar{x})^2 \right)^{\frac{1}{2}} \quad \textbf{(1)}$$

Entropy is a statistical measure of randomness and is defined in Equation 2 where p contains the histogram counts.

$$Entropy = -sum(p \times \log_2 p) \quad \textbf{(2)}$$

The function in MATLAB normalizes the GLCM ($p[i,j]$) in order for the sum of its element (i,j) to be equal to 1. The element specifies the number of times that the pixel with value i occurred horizontally adjacent to a pixel with value$j$. Equation 3 shows the formula for contrast.

$$Contrast = \sum_{i,j} |i-j|^2 p(i,j) \quad \textbf{(3)}$$

Equation 4 shows the formula for correlation where μ is the mean and σ is the standard deviation.

$$Correlation = \sum_{i,j} \frac{(i-\mu i)(j-\mu j)p(i,j)}{\sigma_i \sigma_j} \quad \textbf{(4)}$$

Equation 5 is the formula for energy while equation 6 shows the formula for homogeneity.

$$Energy = \sum_{i,j} p(i,j)^2 \quad \textbf{(5)}$$

$$Homogeneity = \sum_{i,j} \frac{p(i,j)}{1+|i-j|} \quad \textbf{(6)}$$

3.2.6 Pattern Classification

The output parameters would be the five staining pattern which are homogeneous, nucleolar, coarse speckled, fine speckled and centromere. The outcome

of this project is to assess the classification algorithm by validating the accuracy of the pattern classification using Fuzzy Logic by using the fuzzy logic toolbox in MATLAB.

3.3 Project Tasks

i. Literature review

Extensive research and literature review is carried out once the project title is confirmed and the scope of study is identified. This is done to acquire an in-depth understanding on the project.

ii. Justify choice of techniques

Based on the literature review, techniques to extract the textural features and classification are finalized.

iii. Learn MATLAB programming

To be familiar with the ways of coding platform by using online sources and reference books.

iv. Compile coding to M-file

Lines of coding are compiled to M-file before simulation is executed.

v. Run simulation

The algorithm developed in MATLAB is run to obtain data for analysis.

vi. Test feature extraction coding

To verify and debug the coding in feature extraction phase.

vii. Test the classifier

To assess the performance of fuzzy logic as classifier

viii. Improve algorithm

Further improvements are implemented after testing to increase the accuracy of pattern recognition algorithm.

3.4 Gantt Chart and Milestones

Please refer to the attached document in Appendix A.

3.5 Software

Table 5 shows the software utilized in order to complete the project.

Table 5: Softwares

Software	Details	Function
MATLAB R2012a	High performance language for technical computing	Perform image processing from the initial stage of pre-processing to the final classification stage
Fuzzy Logic Toolbox	A graphical user interface (GUI) with many functions that provides an environment for fuzzy interference system design, analysis and implementation	Create and edit fuzzy inference system by inserting the extracted textural features as input and the five staining patterns as output
Microsoft Excel 2010	Electronic spreadsheet program that is used to store data	Store numerical data obtained from feature extraction and calculate the average value of each features to obtain the feature vector of each pattern

3.6 Project Schedule

The planned schedule for Final Year Project I and Final Year Project II are displayed in Table 6 and Table 7.

Table 6: Project schedule for Final Year Project I

Title selection	20 May – 31 May
Extended proposal submission	28 June
Proposal defense	8 July – 12 July
Draft interim report	15 August
Final interim report	23 August

Table 7: Project schedule for Final Year Project II

Progress Report	11 November
ElectrEx	4 December
Draft Report	16 December
Final Report	23 December
Technical Paper	23 December
VIVA	31 December
Hard bound Report	6 January

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Pre-processing and Feature Extraction

Before extracting the features, the HEp-2 images are converted from RGB image to greyscale images. Figure 6 shows an example of the original image and the greyscale image.

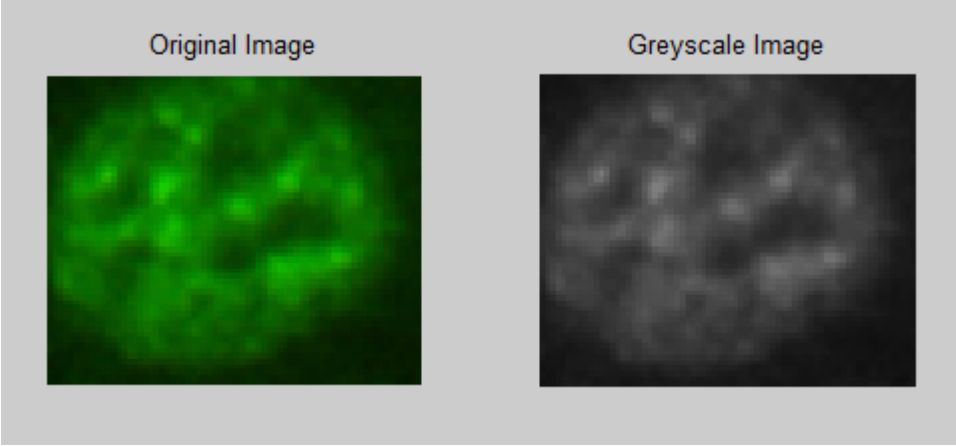


Figure 6: Original and greyscale image

The features are extracted from 425 HEp-2 cell images where using the ratio of 70% training set and 30% testing set, results in 300 images for training and 125 images for testing. Thus, each pattern has 60 images for training and 25 images for testing.

The MATLAB code for the feature extraction from first-order statistics and second-order statistics by using GLCM can be referred in Appendix B. From the coding, seven features are extracted; mean, standard deviation, entropy, contrast, correlation, energy and homogeneity. For GLCM, the offset values are set to be [0 D;-D D; -D 0; -D –D] where D is the pixel distance and is set to 1. This parameter is changed to accommodate different angles of pixel of interest and its neighbour and corresponds to the value of 0°, 45°, 90° and 135° respectively. The obtained numerical value once the simulation is run is exported to Microsoft Excel to ease the analysis of each pattern textural features and to calculate the average value of each feature in order to obtain the

feature vector of each pattern. Appendix C shows the features extracted from the training set. Table 8 shows the average value derived from each pattern.

Table 8: Average value derived from each pattern

Patterns	Mean	Standard Deviation	Entropy	Contrast	Correlation	Energy	Homogeneity
Homogeneous	31.4984	8.2566	4.6919	0.0417	0.7797	0.7559	0.9791
Fine Speckled	36.9177	10.6313	5.1192	0.0398	0.7045	0.7511	0.9801
Coarse Speckled	37.3351	12.6226	5.4058	0.0652	0.7995	0.6175	0.9674
Centromere	20.7766	11.3934	4.8719	0.0900	0.8302	0.4668	0.9551
Nucleolar	23.9000	6.1898	4.2068	0.0367	0.6568	0.7977	0.9817

The feature vector for each pattern is as follows where each column represents the textural features according to Table 8.

Homogenous = [31.4984 8.2566 4.6919 0.0417 0.7797 0.7559 0.9791]

Fine Speckled = [36.9177 10.6313 5.1192 0.0398 0.7045 0.7511 0.9801]

Coarse Speckled = [37.3351 12.6226 5.4058 0.0652 0.7995 0.6175 0.9674]

Centromere = [20.7766 11.3934 4.8719 0.0900 0.8302 0.4668 0.9551]

Nucleolar = [23.9000 6.1898 4.2068 0.0367 0.6568 0.7977 0.9817]

4.2 Pattern Classification

Pattern classification is executed by using the Fuzzy Logic Toolbox in MATLAB. The Mamdani fuzzy inference system is chosen as the output for this classification is neither linear nor constant. The seven textural features are set as the input while the five staining patterns are set as the output as shown in Figure 7.

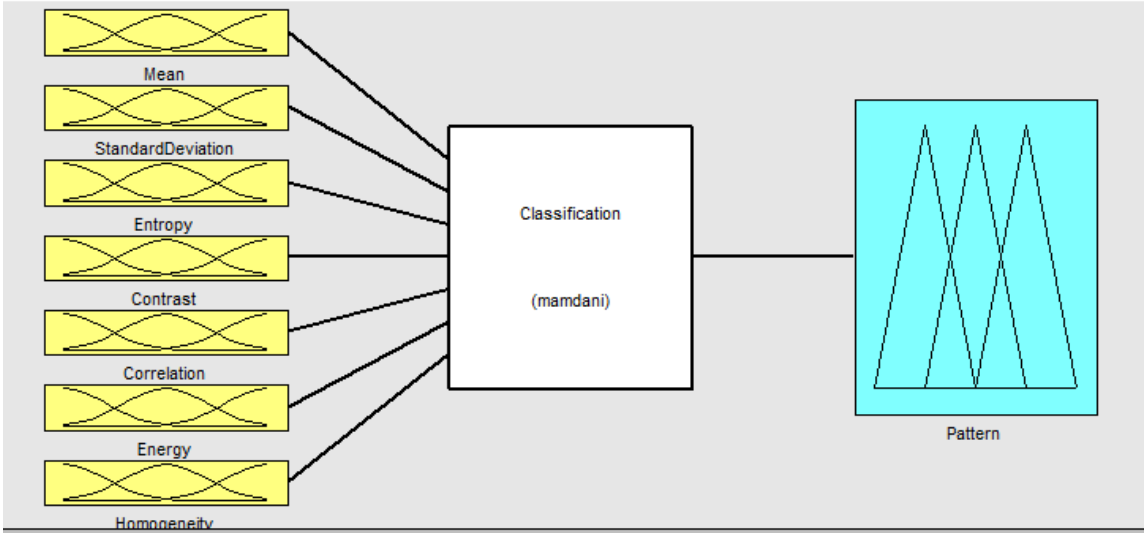


Figure 7: Input and output to the fuzzy inference system

The feature vector obtained previously will now be used as parameters for each of the inputs membership function. Each input would have five membership functions to correspond to the value of each feature for each pattern. The shape of the input membership function used is Gaussian membership function due to their smoothness, crisp notation and being nonzero at all points. Figure 8 shows an example where the input variable is energy and it has 5 membership functions to represent each staining pattern.

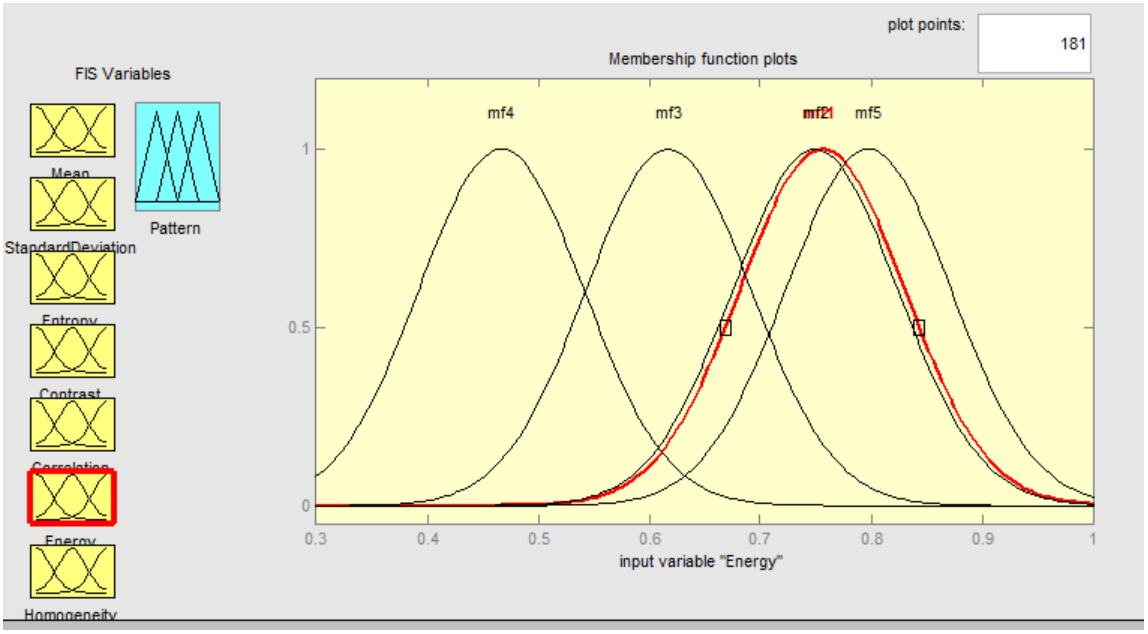


Figure 8: Energy as input variable

Table 9 summarizes the membership functions for input variables.

Table 9: Membership functions for input variables

Pattern	Mean	Standard Deviation	Entropy	Contrast	Correlation	Energy	Homogeneity
Homogenous	mf1	mf1	mf1	mf1	mf1	mf1	mf1
Fine Speckled	mf2	mf2	mf2	mf2	mf2	mf2	mf2
Coarse Speckled	mf3	mf3	mf3	mf3	mf3	mf3	mf3
Centromere	mf4	mf4	mf4	mf4	mf4	mf4	mf4
Nucleolar	mf5	mf5	mf5	mf5	mf5	mf5	mf5

Creation of the membership function for the output variables is done in a similar manner where it has 5 membership functions to represent each pattern. The shape of the output membership functions are triangular membership function. It is a straight line membership function and has the advantage of simplicity. The selection of the membership functions is wide but uncommon membership functions are not required for a good fuzzy inference system. Figure 9 shows the output membership functions while Table 10 shows the parameter values for the output variables.

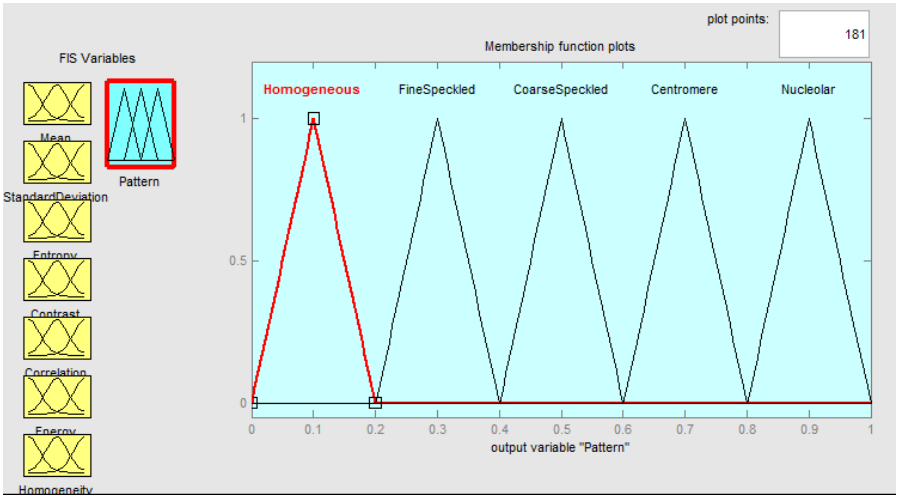


Figure 9: Output membership functions

Table 10: Parameter values for output variables

Patterns	Parameter
Homogenous	0 – 0.2
Fine Speckled	0.201 – 0.4
Coarse Speckled	0.401 – 0.6
Centromere	0.601 – 0.8
Nucleolar	0.801 – 1.0

Once the variables have been named and suitable shapes are set for the membership functions, the rule-based structure of fuzzy logic is implemented as shown in Figure 10.

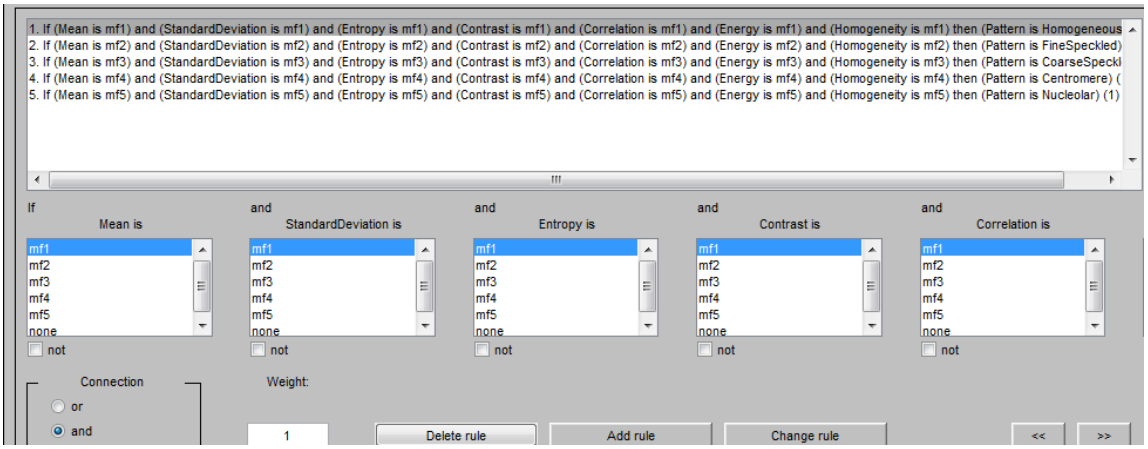


Figure 10: Rule-based structure for fuzzy logic

At this point, the fuzzy inference system is completely defined. The variables, membership functions and rules construction allows the system to calculate for the classification stage. The feature vector of each pattern from the testing set is used as the input parameters in the rule viewer. Figure 11 shows the rule viewer where the input value is a feature vector for centromere staining pattern. It is observed that for this particular example, fuzzy logic is able to classify the pattern as centromere as the output value falls between the ranges of 0.601 to 0.8. The classification results and the feature vector for testing set can be found in Appendix D.

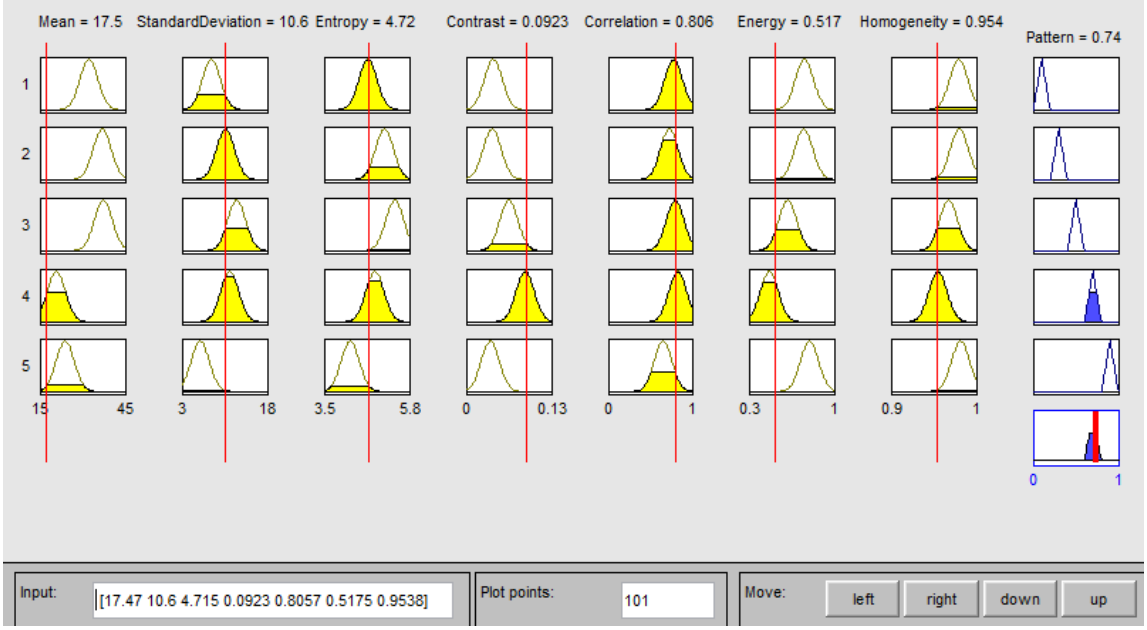


Figure 11: Rule viewer

Each column of plots in yellow shows how the input variable is used in the rules with the input values shown on top. The column of plots in blue shows how the output variable is used in the rules. The bottom-right plot shows how the output of each rule is combined to make an aggregate output and then defuzzified. The thick red line in the bottom- right plot provides the defuzzified value. The defuzzification method used is largest of maximum which key off the maximum value assumed by the aggregate membership function.

This method is run for all images in the testing set. Table 11 shows the classification results and rate of accuracy while Figure 12 shows a graph of the classification results.

Table 11: Classification results

Pattern	Total image	Classified	Misclassified	Accuracy (%)
Homogenous	25	19	6	76
Fine speckled	25	20	5	80
Coarse Speckled	25	21	4	84
Centromere	25	24	1	96
Nucleolar	25	21	4	84
			Mean Accuracy (%)	**84**

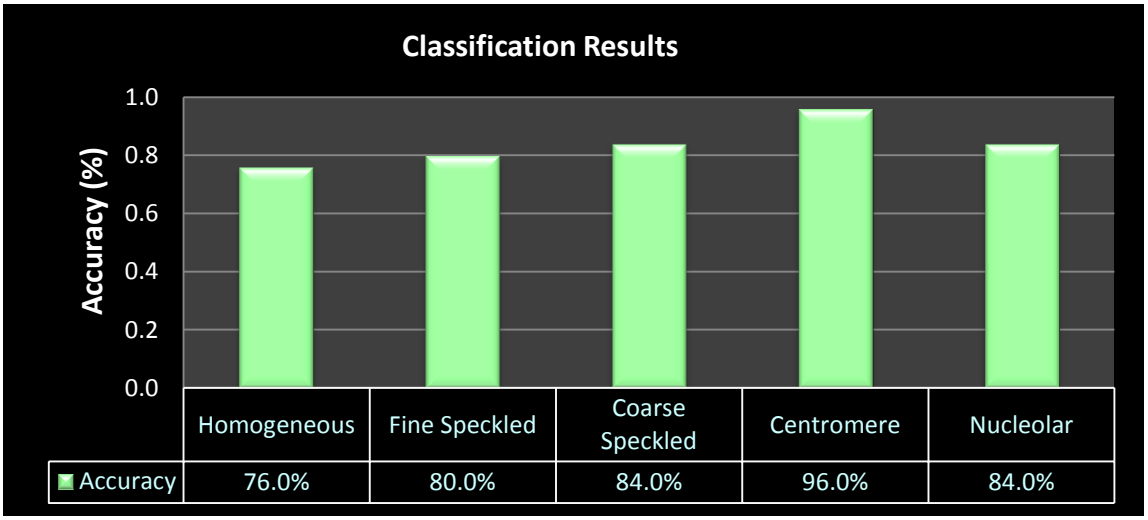


Figure 12: Graph of classification results

From Table 11, it can be observed fuzzy logic as a classifier for HEp-2 cell pattern recognition gives a mean accuracy of 84%.

The misclassification of pattern might be a result of overlapping textural feature values. The staining pattern has similarity when being analysed visually. This is also proven through the textural features extraction as it can be observed that the features values are not completely distinct between each pattern which makes classification more complex. Patterns sharing the same visual appearance such as nucleolar and centromere are usually confused with each other. The same remark can be made for the remaining three patterns.

4.3 Graphical User Interface (GUI)

A GUI as shown in Figure 13 is created using MATLAB as well to increase the efficiency and enhance the ease of use of the automated pattern classification system. The GUI allows user to select an image from the dataset by clicking on the pushbutton. It will then show the original image and greyscale image. A table below the images gives information on the value of each textural features and the box at the bottom will display the staining pattern. The coding for the GUI can be found in Appendix E.

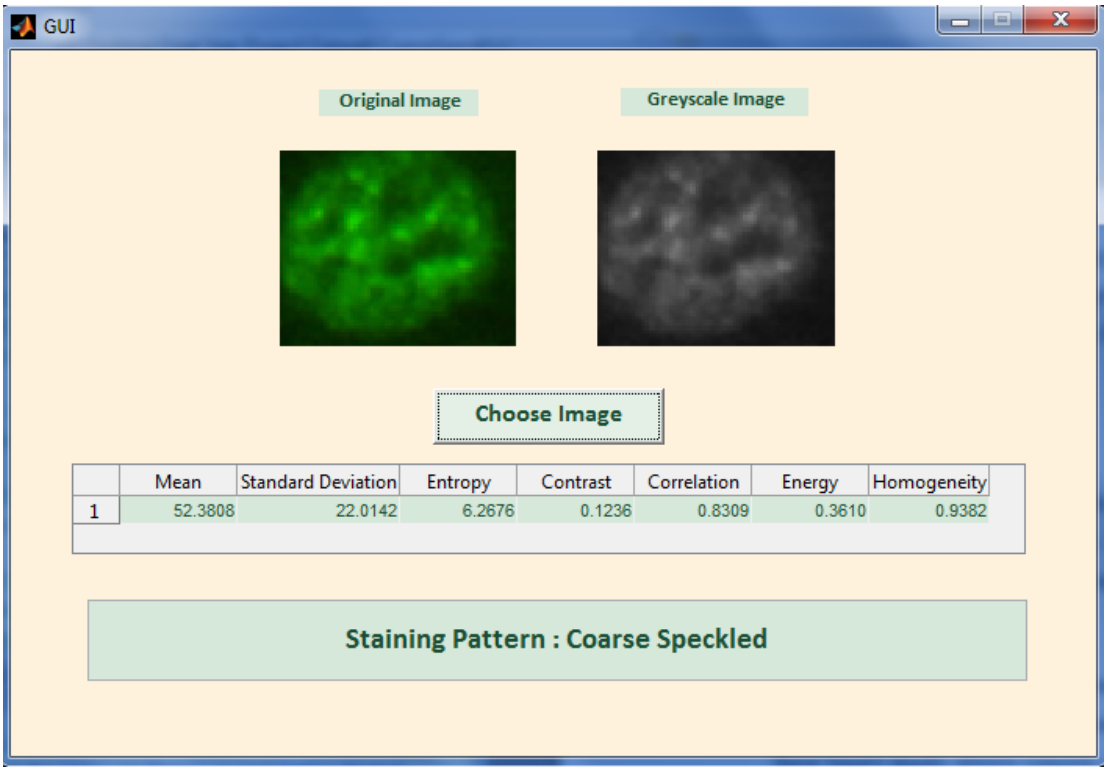


Figure 13: Graphical User Interface

CHAPTER 5
CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

An autoimmune disorder occurs when a person's immune system mistakenly attacks their own body. There are many different autoimmune diseases ranging in different severity; from mild to disabling which depends on which system of the body is under attack and to what degree. Manual visual analysis by medical specialists is a time-consuming operation. Besides, the results produced by the experts are ambiguous at times as it depends on subjective factors such as years of experience of the medical specialists. Hence, an automated pattern classification system plays an important role to reduce the reliance on the experts while maintaining the quality and decreasing the time for diagnosis. Early diagnosis is also necessary as autoimmune diseases are chronic conditions with no cure and treatment involves attempts to control the process of disease and to decrease the symptoms.

Based on the research that has been conducted, statistical approaches are more suitable to extract textural features as HEp-2 cell images have small basic patterns. Statistical approaches selected for this project are first order statistics and second order statistics based on GLCM. Seven textural features are extracted in total which are mean, standard deviation, entropy, contrast, correlation, energy and homogeneity. The HEp-2 cell image is converted to greyscale and the average values of the features are calculated from the training dataset. The average values of the seven features are then used as the input variable for the fuzzy logic classifier. Mamdani fuzzy inference system is used with gaussian membership function for the input and triangle membership function as the output. The defuzzification method utilized is the largest of maximum. The results generated from the proposed method enables the classification of five staining patterns.

The algorithm of this study successfully classifies five staining patterns of the HEp-2 cell images classification by using statistical texture analysis and fuzzy logic, hence, fulfilling the objectives of the project. The algorithm is able to produce a mean accuracy of 84% out of 125 test images. This shows that the algorithm has development potential due to the increasing need of automated classification system in the field of clinical pathology tests.

5.2 Recommendations

There are two recommendations that could be implemented to further improve the project. The first recommendation involves developing an automated segmentation algorithm. The images obtained from MIVIA dataset has been manually segmented to cell level from the image level. Due to time constraint, the author utilizes the cell images which have been segmented for the project. An automated segmentation algorithm would be highly recommended as images obtained from diagnostic centers have numerous HEp-2 cells in an image. It is necessary to extract each single cell from the image for further image processing and analysis.

Further work on this project could also involve other statistical approaches to extract the textural features from the images. Statistical approaches are not limited to just first-order statistics and GLCM. Combination of other techniques to be used as a descriptor could possibly increase the accuracy of the classification. Examples of other statistical methods that could be considered are Law's texture energy measures which determines texture properties by assessing average grey level, edges, spot, ripples as well as waves and edge frequency which measures the total length of all edges in a region to measure coarseness or complexity of a texture.

REFERENCES

[1] P. Elbischger, S. Geerts, K. Sander, G. Ziervogel-Lukas, and P. Sinah, "Algorithmic framework for HEp-2 fluorescence pattern classification to aid auto-immune diseases diagnosis," in *Biomedical Imaging: From Nano to Macro, 2009. ISBI'09. IEEE International Symposium* on, Oakland,California, 2009, pp. 562-565.

[2] S. Di Cataldo, A. Bottino, E. Ficarra, and E. Macii, "Applying textural features to the classification of HEp-2 cell patterns in IIF images," in *Pattern Recognition (ICPR), 2012 21st International Conference* on, Tsukuba,Japan, 2012, pp. 3349-3352.

[3] G. Iannello, L. Onofri, and P. Soda, "A bag of visual words approach for centromere and cytoplasmic staining pattern classification on HEp-2 images," in *Computer-Based Medical Systems (CBMS), 2012 25th International Symposium* on, Rome,Italy, 2012, pp. 1-6.

[4] MIVIA HEp-2 Images Dataset, 2010. [Online].Available: http://mivia.unisa.it/datasets/biomedical-image-datasets/hep-2-image-dataset/

[5] V. Snell, W. Christmas, and J. Kittler, "Texture and shape in fluorescence pattern identification for auto-immune disease diagnosis," *in Pattern Recognition (ICPR), 2012 21st International Conference* on, Tsukuba,Japan, 2012, pp. 3750-3753.

[6] R. V. Rajesh, J. Veerappan, S. K. Sujitha, and E. A. Kumar, "Classification and retrieval of images using texture features," in *Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on*, Coimbatore,India, 2012, pp. 1-5.

[7] N. Sharma, A. K. Ray, S. Sharma, K. Shukla, S. Pradhan, and L. M. Aggarwal, "Segmentation and classification of medical images using texture-primitive features: Application of BAM-type artificial neural network," *Journal of medical physics/Association of Medical Physicists of India*, vol. 33, p. 119, 2008.

[8] R. C. Jain, R. Kasturi, and B. G. A. SCHUNCK, *Machine Vision*. New York: McGraw-Hill Higher Education, 1995. *of Medical Physicists of India*, vol. 33, p. 119, 2008.

[9] E. Sivasankar and R. Rajesh, "Knowledge discovery in medical datasets using a fuzzy logic rule based classifier," in *Electronic Computer Technology (ICECT), 2010 International Conference* on, Kuala Lumpur,Malaysia, 2010, pp. 208-213.

[10] B. Pham, S. Albarede, A. Guyard, E. Burg, and P. Maisonneuve, "Impact of external quality assessment on antinuclear antibody detection performance," *Lupus*, vol. 14, pp. 113-119, 2005.

[11] R. Hiemann, T. Büttner, T. Krieger, D. Roggenbuck, U. Sack, and K. Conrad,"Challenges of automated screening and differentiation of non-organ specific autoantibodies on HEp-2 cells," *Autoimmunity Reviews*, vol. 9, pp. 17-22, 2009.

[12] A. Rigon, P. Soda, D. Zennaro, G. Iannello, and A. Afeltra, "Indirect immunofluorescence in autoimmune diseases: assessment of digital images for diagnostic purpose," *Cytometry Part B: Clinical Cytometry*, vol. 72, pp. 472-477, 2007.

[13] A. Khanna and M. Shrivastava, "Unsupervised techniques of segmentation on texture images: A comparison," in *Signal Processing, Computing and Control (ISPCC), 2012 IEEE International Conference* on, 2012, pp. 1-6.

[14] P. Strandmark, J. Ulén, and F. Kahl, "Hep-2 staining pattern classification," in *Pattern Recognition (ICPR), 2012 21st International Conference* on, Tsukuba,Japan, 2012, pp. 33-36.

[15] S. Xiaoyi, L. Yongjie, and C. Wufan, "A Textural Feature-Based Image Retrieval Algorithm," in *Natural Computation, 2008. ICNC '08. Fourth International Conference* on, Jinan,China, 2008, pp. 71-75.

[16] M. Amadasun and R. King, "Textural features corresponding to textural properties," *Systems, Man and Cybernetics, IEEE Transactions* on, vol. 19, pp. 1264-1274, 1989.

[17] J. R. Jang, MATLAB: *Fuzzy Logic Toolbox User's Guide: Version 1*: Math Works, 1997.

[18] G. Srinivasan and G. Shobha, "Statistical texture analysis," *proceedings of world academy of science, engg & tech*, vol. 36, 2008.

[19] A. Materka and M. Strzelecki, "Texture analysis methods-a review," *Technical university of lodz, institute of electronics*, COST B11 report, Brussels, pp. 9-11, 1998.

[20] R. Maini and H. Aggarwal, "Study and comparison of various image edge detection techniques," *International Journal of Image Processing (IJIP)*, vol. 3, pp. 1-11, 2009.

[21] P. Foggia, G. Percannella, P. Soda, and M. Vento, "Benchmarking HEp-2 Cells Classification Methods," *Medical Imaging, IEEE Transactions* on, vol. PP, pp. 1-1, 2013.

[22] K. Li, J. Yin, Z. Lu, X. Kong, R. Zhang, and W. Liu, "Multiclass boosting SVM using different texture features in HEp-2 cell staining pattern classification," in *Pattern Recognition (ICPR), 2012 21st International Conference* on, 2012, pp. 170-173.

[23] R. Nosaka, Y. Ohkawa, and K. Fukui, "Feature extraction based on co-occurrence of adjacent local binary patterns," in *Advances in Image and Video Technology*, ed: Springer, 2012, pp. 82-91.

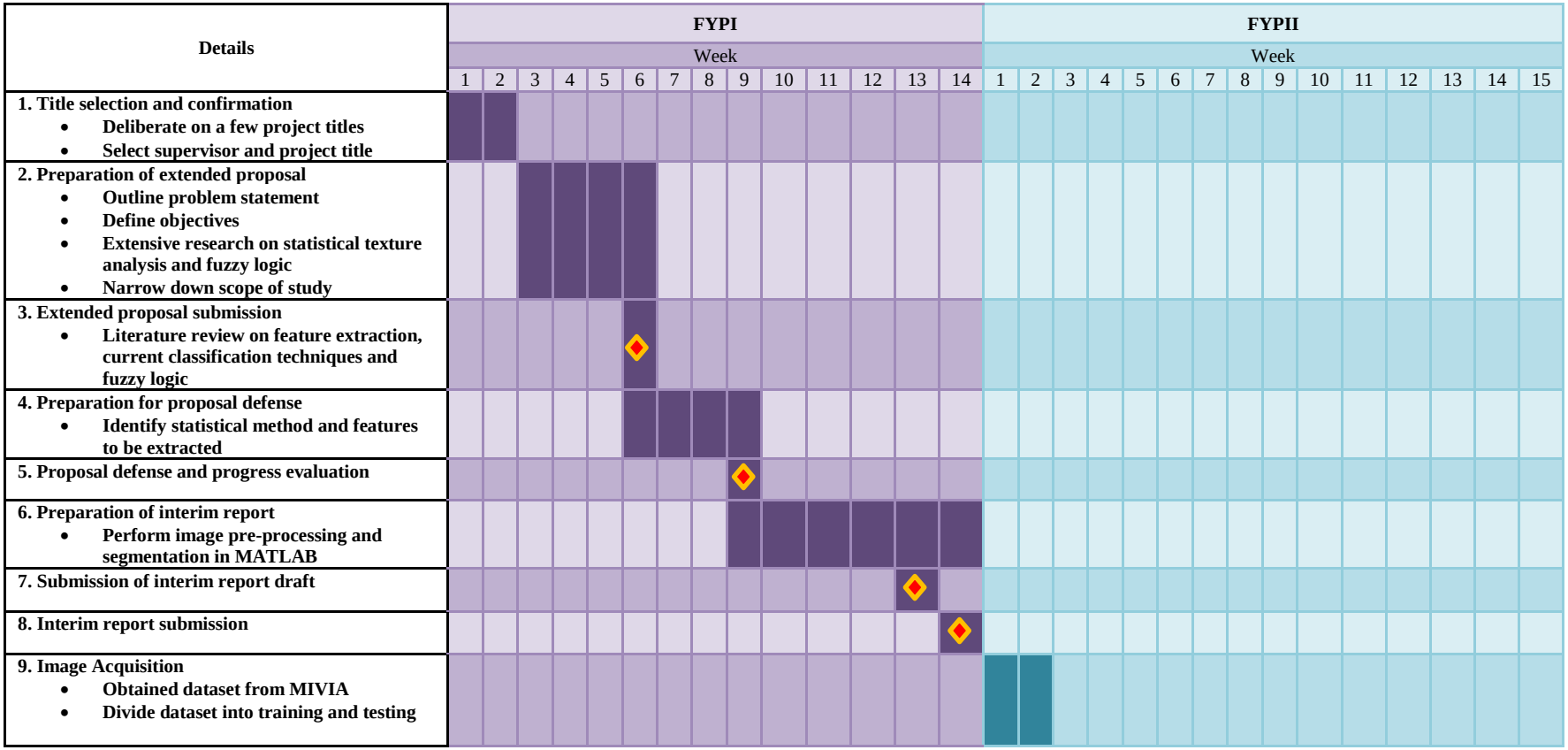
[24] X. Kong, K. Li, Z. Lu, and L. Wenyin, "HEp-2 cell classification using a new texture descriptor," presented at the *Pattern Recognition (ICPR), 2012 21st International Conference* on, 2012, Tsukuba,Japan, 2012.

[25] G. Nai Ren, K. Chao-Lin, T. Tzong-Jiy, and C. Shi-Jaw, "An adaptive fuzzy classification system," in *Soft Computing in Industrial Applications, 2008. SMCia '08. IEEE Conference* on Muroran, Japan, 2008, pp. 377-381.

[26] G. Schaefer, T. Nakashima, M. Zavisek, Y. Yokota, A. Drastich, and H. Ishibuchi, "Breast cancer classification using statistical features and fuzzy classification of thermograms," in *Fuzzy Systems Conference, 2007. FUZZ-IEEE 2007. IEEE International*, London, United Kingdom, 2007, pp. 1-5.

APPENDICES

APPENDIX A: GANTT CHART AND MILESTONES



Details	FYPI														FYPII													
	Week														Week													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
10. Feature Extraction • **4 features from GLCM** • **3 features from first order statistics** • **Write MATLAB code, run and verify**																													
11. Classification • **Classify image into 5 patterns using Fuzzy Logic Toolbox in MATLAB by inserting 7 features as input** • **Verify results**																													
12. Submission of Progress Report • **Update progress from FYPI**																													
13. Preparation for ElectrEx • **Compile all the findings** • **Design poster**																													
14. Participate in ElectrEx • **Poster presentation** • **Present project progress/results**																													
15. Preparation of Final Report • **Include final result/findings** • **Accuracy rate of results** • **Note down any recommendations for future improvement if any**																													
16. Preparation of Technical Paper																													
17. Submission of Final Report																													
18. Submission of Technical Paper																													
19. Preparation for project VIVA																													
19. Project VIVA																													

Key milestone Suggested milestone

APPENDIX B: CODING FOR FEATURE EXTRACTION AND PATTERN CLASSIFICATION

```
clear all;
clc;
features = [];
%% PREPROCESSING

NF = 129;               %choose the number of images as input
prefix_image='';        %change the desired input image name
fileformat='.png';      %change the desired input image

for num=1:NF
  I = imread(strcat(prefix_image,num2str(num),fileformat));
  ImageGrey = rgb2gray(I);
  %% %% First-order statistics


A=double(ImageGrey);

Mean = mean(ImageGrey(:));
Variance = var(A(:));
StdDev = sqrt(Variance(:));
Entropy = entropy(ImageGrey);

%% GLCM

offsets1 = [0 1;-1 1;-1 0;-1 -1];
glcm = graycomatrix (ImageGrey, 'GrayLimits', [0 255], 'NumLevels', 8, 'Offset',
offsets1, 'Symmetric', true);
stats = graycoprops (glcm, 'contrast');
stats2 = graycoprops (glcm, 'correlation');
stats3 = graycoprops (glcm, 'energy');
stats4 = graycoprops (glcm, 'homogeneity');

t = struct2array(stats);
t2 = struct2array(stats2);
t3 = struct2array(stats3);
t4 = struct2array(stats4);

Contrast = mean(t);
Correlation = mean(t2);
Energy = mean(t3);
Homogeneity = mean(t4);

%% Saving Data

features = [features; Mean StdDev Entropy Contrast Correlation Energy Homogeneity];
%% Fuzzy Logic

fismat = readfis('Classification');
out = evalfis([Mean StdDev Entropy Contrast Correlation Energy Homogeneity],fismat);
fprintf('%d\t Output = %0.3f \n',num,out);

    if out<0.201
        disp('The staining pattern is homogeneous')
    elseif out<0.401
        disp('The staining pattern is fine speckled')
    elseif out<0.601
        disp('The staining pattern is coarse speckled')
    elseif out<0.801
        disp('The staining pattern is centromere')
    else
        disp('The staining pattern is nucleolar')
    end
end
```

APPENDIX C: FEATURES EXTRACTED FROM TRAINING SET

Mean	Std Dev	Entropy	Contrast	Correlation	Energy	Homogeneity	Pattern
41.3824	15.0913	5.6890	0.0645	0.8761	0.5152	0.9678	Homogeneous
42.4993	12.4711	5.3314	0.0863	0.7708	0.5928	0.9568	Homogeneous
41.0748	12.3839	5.0326	0.0807	0.7495	0.6466	0.9597	Homogeneous
33.6851	8.7036	4.8289	0.0422	0.5675	0.8635	0.9789	Homogeneous
35.2776	10.5721	4.9211	0.0613	0.6029	0.7903	0.9694	Homogeneous
38.0645	10.5198	4.9333	0.0456	0.5917	0.8474	0.9772	Homogeneous
36.8033	11.7244	5.3896	0.0676	0.7366	0.7071	0.9662	Homogeneous
38.8336	10.3870	4.7860	0.0402	0.7748	0.7903	0.9799	Homogeneous
43.0133	14.9298	5.4991	0.1182	0.8007	0.4342	0.9409	Homogeneous
36.8287	10.6253	5.0387	0.0408	0.7321	0.8126	0.9796	Homogeneous
30.6395	9.1750	4.9251	0.0447	0.8072	0.7251	0.9777	Homogeneous
35.4911	10.3675	4.5578	0.0305	0.8846	0.7086	0.9848	Homogeneous
33.9128	8.1664	4.6330	0.0226	0.8563	0.8203	0.9887	Homogeneous
35.2386	9.4709	5.1664	0.0224	0.7621	0.8841	0.9888	Homogeneous
26.3815	6.2460	4.4687	0.0357	0.8580	0.7139	0.9822	Homogeneous
38.0432	13.0085	5.3949	0.0696	0.7940	0.6423	0.9652	Homogeneous
29.7077	8.4954	4.9794	0.0286	0.7879	0.8405	0.9857	Homogeneous
30.7518	8.6153	4.7128	0.0224	0.9098	0.7297	0.9888	Homogeneous
29.6227	6.7297	4.7184	0.0307	0.7654	0.8391	0.9847	Homogeneous
29.1906	6.4897	4.4715	0.0429	0.7467	0.7891	0.9785	Homogeneous
27.5668	5.7764	4.4747	0.0285	0.7923	0.8344	0.9857	Homogeneous
27.9070	6.9319	4.2791	0.0309	0.8909	0.6866	0.9846	Homogeneous
26.8150	6.2040	4.6111	0.0267	0.8422	0.8048	0.9867	Homogeneous
30.5118	9.6834	5.1833	0.0247	0.8870	0.7584	0.9877	Homogeneous
25.2113	5.9847	4.5276	0.0306	0.8832	0.7084	0.9847	Homogeneous
35.1653	10.7516	4.8497	0.0579	0.6911	0.7577	0.9711	Homogeneous
25.7008	4.6688	4.0938	0.0281	0.8348	0.8019	0.9859	Homogeneous
28.6095	6.4484	4.2351	0.0331	0.8606	0.7302	0.9835	Homogeneous
38.8061	10.8747	4.6601	0.0463	0.6906	0.8071	0.9769	Homogeneous
40.5919	13.8095	5.6207	0.0793	0.7680	0.6215	0.9604	Homogeneous
33.5494	9.4939	4.9326	0.0327	0.7975	0.8066	0.9837	Homogeneous
44.5784	15.0576	5.6421	0.0927	0.8509	0.4173	0.9537	Homogeneous
29.5067	5.1133	4.3226	0.0142	0.7288	0.9338	0.9929	Homogeneous
32.2657	8.8115	4.9141	0.0307	0.8050	0.8124	0.9846	Homogeneous
36.9344	11.2006	4.8535	0.0558	0.6854	0.7715	0.9721	Homogeneous
29.1948	6.4386	4.3961	0.0392	0.7767	0.7863	0.9804	Homogeneous
31.0858	7.3719	4.5508	0.0503	0.5875	0.8301	0.9749	Homogeneous
29.2627	7.6811	4.7299	0.0566	0.7306	0.7364	0.9717	Homogeneous
31.7891	5.9625	4.5696	0.0058	0.7947	0.9657	0.9971	Homogeneous
30.6973	5.6396	4.4806	0.0107	0.4156	0.9709	0.9946	Homogeneous
34.0723	9.5831	4.8606	0.0322	0.8083	0.8006	0.9839	Homogeneous
27.2588	3.7133	3.8471	0.0110	0.7746	0.9399	0.9945	Homogeneous
34.3082	8.8593	4.8892	0.0217	0.8783	0.8001	0.9892	Homogeneous
32.9174	9.4343	4.5769	0.0356	0.8617	0.7074	0.9822	Homogeneous
27.2815	6.6397	4.4622	0.0402	0.8565	0.6804	0.9799	Homogeneous
28.1333	6.4368	4.3719	0.0390	0.8254	0.7383	0.9805	Homogeneous
26.7132	6.3347	4.4374	0.0411	0.8467	0.6919	0.9794	Homogeneous
26.1275	5.6885	4.4306	0.0443	0.7875	0.7487	0.9779	Homogeneous
34.5963	11.5067	5.3958	0.0596	0.7549	0.7248	0.9702	Homogeneous
25.8784	5.2486	4.2108	0.0294	0.8457	0.7807	0.9853	Homogeneous

25.3565	6.0954	4.4546	0.0390	0.8705	0.6611	0.9805	Homogeneous
25.7985	5.4953	4.4091	0.0362	0.8178	0.7655	0.9819	Homogeneous
24.5389	5.0430	4.0595	0.0402	0.8594	0.6747	0.9799	Homogeneous
22.8554	4.3786	4.0487	0.0466	0.8458	0.6528	0.9767	Homogeneous
23.2339	3.2988	3.6887	0.0414	0.7563	0.7902	0.9793	Homogeneous
24.4431	5.5241	4.3492	0.0536	0.7985	0.6824	0.9732	Homogeneous
24.4809	4.8848	4.2251	0.0486	0.7670	0.7452	0.9757	Homogeneous
25.0799	6.2187	4.5281	0.0287	0.8964	0.6950	0.9857	Homogeneous
29.5426	6.9008	4.6808	0.0161	0.6201	0.9416	0.9919	Homogeneous
25.0980	6.0357	4.1853	0.0275	0.6201	0.9007	0.9863	Homogeneous
55.2536	10.9202	5.2772	0.0690	0.8659	0.4335	0.9655	Fine Speckled
43.6037	12.5152	5.3618	0.0783	0.8046	0.5783	0.9608	Fine Speckled
43.7281	11.9392	5.1143	0.0898	0.7096	0.6372	0.9551	Fine Speckled
40.4127	15.0732	5.5193	0.0590	0.8948	0.5087	0.9705	Fine Speckled
48.0057	17.4275	5.8810	0.0388	0.9471	0.4101	0.9806	Fine Speckled
33.5521	12.1927	5.3827	0.0402	0.8678	0.6648	0.9799	Fine Speckled
30.2129	12.6322	5.3667	0.0514	0.8794	0.5390	0.9743	Fine Speckled
31.5815	12.6123	5.5029	0.0403	0.8959	0.6128	0.9799	Fine Speckled
34.5768	12.8002	5.5100	0.0410	0.8808	0.6484	0.9795	Fine Speckled
31.8291	11.1154	5.3137	0.0370	0.8647	0.6921	0.9815	Fine Speckled
35.1665	11.4517	5.2237	0.0421	0.7978	0.7572	0.9790	Fine Speckled
37.2693	13.3035	5.4846	0.0645	0.8249	0.6227	0.9678	Fine Speckled
33.3262	12.2133	5.2993	0.0329	0.8935	0.6613	0.9836	Fine Speckled
41.0707	12.7603	5.4442	0.0767	0.7913	0.6144	0.9616	Fine Speckled
37.0601	11.4944	5.3150	0.0551	0.6733	0.7888	0.9725	Fine Speckled
35.3370	12.8348	5.4053	0.0643	0.8090	0.6369	0.9679	Fine Speckled
39.9288	14.7144	5.6112	0.0749	0.8663	0.4928	0.9626	Fine Speckled
36.7532	12.4066	5.3946	0.0574	0.7909	0.6988	0.9713	Fine Speckled
31.0380	9.4690	5.1482	0.0189	0.8934	0.8039	0.9906	Fine Speckled
37.6414	8.2575	4.9595	0.0138	0.4861	0.9595	0.9931	Fine Speckled
36.0903	13.3491	5.5313	0.0550	0.8300	0.6561	0.9725	Fine Speckled
36.7736	11.4079	5.2606	0.0332	0.8292	0.7812	0.9834	Fine Speckled
36.0694	12.6303	5.3594	0.0477	0.8217	0.7019	0.9762	Fine Speckled
30.7679	10.1151	5.0732	0.0479	0.8401	0.6546	0.9761	Fine Speckled
36.8839	13.0241	5.4597	0.0700	0.8097	0.6209	0.9650	Fine Speckled
39.2193	14.3782	5.5470	0.0916	0.8172	0.5172	0.9542	Fine Speckled
37.4060	13.4055	5.5451	0.0603	0.8314	0.6360	0.9698	Fine Speckled
39.4707	14.3683	5.4857	0.0874	0.8035	0.5493	0.9563	Fine Speckled
37.8845	13.2658	5.4559	0.0607	0.8144	0.6581	0.9697	Fine Speckled
41.1690	14.0636	5.6408	0.0795	0.8369	0.5216	0.9602	Fine Speckled
39.8380	13.0942	5.5067	0.0775	0.7992	0.5999	0.9612	Fine Speckled
34.7242	7.8838	4.8171	0.0000	NaN	1.0000	1.0000	Fine Speckled
38.4832	8.0384	4.8953	0.0041	0.5111	0.9877	0.9980	Fine Speckled
43.2851	8.7076	4.9069	0.0296	0.6885	0.8765	0.9852	Fine Speckled
40.8131	9.6896	4.9858	0.0138	0.4898	0.9597	0.9931	Fine Speckled
34.1216	8.2279	4.7115	0.0057	0.2321	0.9868	0.9971	Fine Speckled
36.1189	8.7963	4.7743	0.0000	NaN	1.0000	1.0000	Fine Speckled
36.1829	8.9437	5.0364	0.0084	0.7376	0.9598	0.9958	Fine Speckled
34.2685	8.3773	4.8367	0.0089	0.6123	0.9679	0.9955	Fine Speckled
28.7365	6.3112	4.3703	0.0184	0.4304	0.9496	0.9908	Fine Speckled
30.4314	5.2795	4.2610	0.0041	0.3342	0.9898	0.9980	Fine Speckled
35.7918	10.3964	5.2112	0.0142	0.8957	0.8506	0.9929	Fine Speckled
32.1999	7.7593	4.6215	0.0055	0.1716	0.9878	0.9972	Fine Speckled
29.1726	5.9706	4.4619	0.0207	0.6594	0.9186	0.9896	Fine Speckled
25.4999	7.2699	4.5594	0.0518	0.8611	0.5782	0.9741	Fine Speckled

26.2159	5.2601	4.0879	0.0582	0.6127	0.7947	0.9709	Fine Speckled
27.7944	4.4592	3.8955	0.0175	0.3287	0.9566	0.9912	Fine Speckled
39.8963	11.9246	5.3742	0.0495	0.7735	0.7431	0.9752	Fine Speckled
38.5359	7.7761	4.7981	0.0002	-0.0001	0.9997	0.9999	Fine Speckled
39.9279	10.9201	5.2686	0.0314	0.7982	0.8141	0.9843	Fine Speckled
45.5334	13.2865	5.5219	0.0485	0.8904	0.5114	0.9757	Fine Speckled
40.1722	13.9732	5.4873	0.0986	0.7790	0.5422	0.9507	Fine Speckled
42.8194	11.9233	5.4317	0.0550	0.8602	0.5858	0.9725	Fine Speckled
45.3181	10.5994	5.2292	0.0521	0.8668	0.5593	0.9739	Fine Speckled
40.1621	9.0477	5.0134	0.0136	0.6376	0.9490	0.9932	Fine Speckled
37.5368	8.3920	4.7822	0.0003	-0.0001	0.9994	0.9999	Fine Speckled
36.9788	7.3394	4.6396	0.0000	NaN	1.0000	1.0000	Fine Speckled
38.4913	6.8848	4.7298	0.0000	NaN	1.0000	1.0000	Fine Speckled
38.0844	7.5525	4.7850	0.0008	0.3746	0.9979	0.9996	Fine Speckled
28.8467	5.6534	4.2808	0.0207	0.4355	0.9431	0.9897	Fine Speckled
52.3808	22.0142	6.2676	0.1236	0.8309	0.3610	0.9382	Coarse Speckled
55.2666	21.8793	6.2954	0.1204	0.8297	0.3625	0.9398	Coarse Speckled
48.2319	20.9999	6.0794	0.1228	0.8221	0.3697	0.9386	Coarse Speckled
42.0117	14.6260	5.7193	0.0957	0.7768	0.5253	0.9521	Coarse Speckled
47.6264	18.5107	6.0314	0.0937	0.8333	0.4247	0.9532	Coarse Speckled
47.4690	18.9903	5.9670	0.0839	0.8409	0.4341	0.9581	Coarse Speckled
47.3973	18.3932	5.9032	0.0803	0.8455	0.4399	0.9599	Coarse Speckled
44.2383	15.3957	5.7769	0.0904	0.7921	0.4973	0.9548	Coarse Speckled
43.0248	14.5747	5.7066	0.0937	0.7531	0.5413	0.9532	Coarse Speckled
44.6836	14.0930	5.6439	0.0833	0.7772	0.5703	0.9583	Coarse Speckled
35.5422	11.2193	5.2465	0.0745	0.6415	0.7369	0.9628	Coarse Speckled
38.0882	12.3885	5.5269	0.0879	0.7094	0.6548	0.9561	Coarse Speckled
43.1336	14.0070	5.6667	0.0966	0.7664	0.5274	0.9517	Coarse Speckled
35.9988	12.1834	5.3945	0.0744	0.7741	0.6393	0.9628	Coarse Speckled
46.5248	19.1326	5.9172	0.1082	0.7934	0.4217	0.9459	Coarse Speckled
46.7218	17.9726	5.9468	0.1051	0.8011	0.4279	0.9474	Coarse Speckled
47.5970	17.4728	5.9546	0.1089	0.7904	0.4224	0.9455	Coarse Speckled
49.0490	17.0670	5.9001	0.0898	0.8219	0.4247	0.9551	Coarse Speckled
49.9067	19.8545	6.0139	0.1294	0.7959	0.3702	0.9353	Coarse Speckled
33.8447	11.2316	5.3203	0.0683	0.7548	0.6771	0.9658	Coarse Speckled
30.4972	9.6229	5.1248	0.0601	0.7648	0.6974	0.9699	Coarse Speckled
31.7378	9.7078	5.1282	0.0533	0.7627	0.7301	0.9734	Coarse Speckled
46.8488	18.9616	6.0145	0.1015	0.8172	0.4168	0.9492	Coarse Speckled
49.6719	16.9408	5.9438	0.1014	0.8020	0.4177	0.9493	Coarse Speckled
48.4548	18.8750	6.0023	0.0997	0.8174	0.4227	0.9502	Coarse Speckled
41.1902	12.9565	5.5437	0.0892	0.6716	0.6519	0.9554	Coarse Speckled
46.6497	15.6620	5.7730	0.0821	0.8252	0.4618	0.9589	Coarse Speckled
47.4425	16.5298	5.8980	0.1058	0.8080	0.4098	0.9471	Coarse Speckled
34.6116	9.1128	5.1319	0.0251	0.5756	0.9175	0.9874	Coarse Speckled
34.2767	9.7661	5.1628	0.0443	0.7512	0.7879	0.9779	Coarse Speckled
29.5858	8.6580	4.9854	0.0520	0.7736	0.7212	0.9740	Coarse Speckled
29.2177	8.5500	4.9338	0.0477	0.8259	0.6804	0.9762	Coarse Speckled
28.4928	7.5589	4.8071	0.0445	0.8015	0.7326	0.9777	Coarse Speckled
24.1972	7.2158	4.6943	0.0454	0.8878	0.5516	0.9773	Coarse Speckled
26.7478	7.7653	4.8529	0.0466	0.8443	0.6557	0.9767	Coarse Speckled
29.0675	9.0670	5.0149	0.0505	0.8320	0.6524	0.9747	Coarse Speckled
28.3006	8.6367	5.0001	0.0509	0.8234	0.6627	0.9745	Coarse Speckled
27.1873	7.9692	4.9142	0.0507	0.8105	0.6837	0.9746	Coarse Speckled
32.9107	9.4487	5.1568	0.0433	0.6917	0.8204	0.9783	Coarse Speckled
28.1195	9.7144	5.0272	0.0472	0.8722	0.5860	0.9764	Coarse Speckled

26.3775	7.9239	4.8593	0.0507	0.8540	0.6043	0.9747	Coarse Speckled
22.8901	5.9962	4.4732	0.0433	0.8908	0.5615	0.9783	Coarse Speckled
24.2489	7.1846	4.7817	0.0648	0.8299	0.5581	0.9676	Coarse Speckled
20.7466	4.7716	4.2220	0.0672	0.8441	0.5057	0.9664	Coarse Speckled
29.7078	8.7991	4.9514	0.0373	0.7759	0.7974	0.9814	Coarse Speckled
34.8643	11.6167	5.4213	0.0371	0.8251	0.7695	0.9814	Coarse Speckled
37.0036	10.6527	5.2679	0.0284	0.8111	0.8225	0.9858	Coarse Speckled
34.7920	16.8919	5.8177	0.0424	0.9363	0.4659	0.9788	Coarse Speckled
34.6791	9.8472	5.2599	0.0217	0.8049	0.8716	0.9891	Coarse Speckled
30.2605	8.2240	4.9861	0.0182	0.8714	0.8474	0.9925	Coarse Speckled
36.7482	17.2865	5.7353	0.0611	0.9257	0.3855	0.9695	Coarse Speckled
28.5528	7.9710	4.9517	0.0189	0.9047	0.7829	0.9906	Coarse Speckled
36.7195	10.6083	5.2937	0.0295	0.8030	0.8307	0.9853	Coarse Speckled
39.0662	11.3105	5.3463	0.0342	0.8076	0.7895	0.9829	Coarse Speckled
36.1076	10.7761	5.3065	0.0296	0.7562	0.8554	0.9852	Coarse Speckled
35.5734	11.8748	5.4222	0.0404	0.8267	0.7499	0.9798	Coarse Speckled
34.2806	11.4390	5.3970	0.0387	0.8388	0.7440	0.9806	Coarse Speckled
34.0876	11.3148	5.2984	0.0401	0.7489	0.8094	0.9799	Coarse Speckled
31.7089	8.6011	5.0238	0.0201	0.8192	0.8688	0.9899	Coarse Speckled
37.7462	9.5423	5.1467	0.0170	0.5847	0.9426	0.9915	Coarse Speckled
28.0752	14.0622	5.4056	0.1089	0.7967	0.4714	0.9457	Centromere
31.4670	15.2497	5.7515	0.0782	0.8504	0.5176	0.9609	Centromere
22.6441	12.6795	5.0788	0.1174	0.8205	0.3789	0.9413	Centromere
19.5667	12.7412	4.9630	0.0809	0.8681	0.4560	0.9597	Centromere
22.9393	13.2445	5.1225	0.1021	0.8456	0.3880	0.9489	Centromere
19.5595	10.9768	4.7093	0.0865	0.8409	0.4633	0.9567	Centromere
20.0931	12.7757	4.7987	0.0977	0.8412	0.4562	0.9514	Centromere
24.4118	16.3407	5.1663	0.1078	0.8708	0.3694	0.9465	Centromere
18.9182	13.0165	4.9907	0.0874	0.8541	0.4763	0.9564	Centromere
18.1664	10.4854	4.8375	0.0984	0.8079	0.4665	0.9508	Centromere
17.2897	10.9460	4.5933	0.0749	0.8430	0.5501	0.9626	Centromere
21.6076	12.3132	5.3009	0.0910	0.8455	0.4043	0.9545	Centromere
16.7490	10.3858	4.6253	0.0744	0.8307	0.5603	0.9628	Centromere
17.7497	10.0675	4.6701	0.0884	0.8183	0.4913	0.9558	Centromere
16.3086	9.9825	4.6237	0.0766	0.8144	0.5731	0.9617	Centromere
20.9627	12.2723	4.7697	0.0826	0.8693	0.4450	0.9587	Centromere
22.8442	13.4143	5.3072	0.0876	0.8636	0.4052	0.9567	Centromere
21.1875	12.4999	5.0478	0.1007	0.8410	0.4075	0.9501	Centromere
24.0233	13.5244	5.0226	0.0969	0.8570	0.3903	0.9518	Centromere
24.4509	15.1950	5.1907	0.1204	0.8342	0.3740	0.9399	Centromere
27.5488	12.7748	5.4327	0.0757	0.8446	0.5145	0.9622	Centromere
33.8484	14.4539	5.5413	0.0942	0.7450	0.6105	0.9533	Centromere
23.1201	13.7658	5.1940	0.1020	0.8480	0.3831	0.9492	Centromere
25.8955	12.9175	5.3345	0.0847	0.8443	0.4589	0.9576	Centromere
23.3140	15.1116	5.2427	0.1084	0.8552	0.3779	0.9461	Centromere
18.9454	11.1619	4.5797	0.0799	0.8489	0.5085	0.9601	Centromere
19.6044	9.6817	4.5205	0.0817	0.8386	0.4783	0.9591	Centromere
20.7601	12.7693	4.8456	0.1046	0.8372	0.4349	0.9480	Centromere
13.4937	8.2449	4.2746	0.0551	0.8119	0.6837	0.9724	Centromere
27.4182	17.3680	5.4979	0.1054	0.8666	0.4005	0.9479	Centromere
31.0559	18.1362	5.7802	0.0906	0.8905	0.3699	0.9548	Centromere
20.1917	11.1360	4.8444	0.1068	0.8151	0.4187	0.9466	Centromere
18.9506	11.2833	4.6604	0.0966	0.8228	0.4710	0.9519	Centromere
14.0503	7.2101	4.0701	0.0506	0.7902	0.7311	0.9747	Centromere
16.0607	7.8882	4.4961	0.0818	0.7912	0.5450	0.9591	Centromere

20.9943	10.6840	4.8778	0.0900	0.8421	0.4176	0.9550	Centromere
17.6528	9.8122	4.7304	0.0772	0.8334	0.5171	0.9614	Centromere
16.2036	7.6646	4.2950	0.0965	0.7350	0.5623	0.9518	Centromere
19.2790	10.3844	4.6821	0.0677	0.8674	0.4850	0.9661	Centromere
23.2076	11.7975	5.0664	0.0988	0.8325	0.4011	0.9506	Centromere
22.7169	11.7495	5.0533	0.1034	0.8211	0.4086	0.9484	Centromere
22.0010	14.3321	5.1236	0.0848	0.8796	0.4065	0.9577	Centromere
20.7023	11.3017	4.6288	0.1007	0.8316	0.4305	0.9498	Centromere
21.8468	10.8160	5.0740	0.1033	0.8153	0.4006	0.9484	Centromere
23.4404	11.3667	5.0839	0.1006	0.8216	0.4098	0.9497	Centromere
19.1069	9.7355	4.5555	0.0967	0.8166	0.4570	0.9516	Centromere
19.1324	11.8814	4.9611	0.0805	0.8552	0.4541	0.9600	Centromere
18.9793	11.8799	4.8428	0.0825	0.8479	0.4801	0.9591	Centromere
16.8480	8.2998	4.3581	0.0817	0.7974	0.5363	0.9592	Centromere
16.5987	7.2862	4.4754	0.0931	0.7682	0.5178	0.9534	Centromere
16.3036	7.5989	4.4617	0.0789	0.8120	0.5107	0.9605	Centromere
16.2285	7.5153	4.4370	0.0873	0.7864	0.5252	0.9563	Centromere
22.0448	11.2346	5.0281	0.0733	0.8741	0.4217	0.9634	Centromere
18.1764	8.8498	4.6195	0.0847	0.8200	0.4790	0.9576	Centromere
17.7188	9.8127	4.6608	0.0802	0.8331	0.4988	0.9599	Centromere
21.1313	10.2653	4.8532	0.0861	0.8475	0.4174	0.9569	Centromere
18.1539	8.3024	4.6000	0.0790	0.8366	0.4605	0.9605	Centromere
19.7600	8.5093	4.6917	0.1076	0.7911	0.4106	0.9462	Centromere
18.9783	8.8989	4.5800	0.1100	0.7749	0.4400	0.9450	Centromere
16.1163	7.5491	4.2845	0.0786	0.8060	0.5276	0.9607	Centromere
18.3332	10.1737	4.4073	0.0509	0.8656	0.5867	0.9745	Nucleolar
19.0629	11.2423	4.5950	0.0745	0.8474	0.5387	0.9627	Nucleolar
19.7227	10.7186	4.7068	0.0781	0.8340	0.4784	0.9609	Nucleolar
19.8763	11.0933	4.6526	0.0682	0.8558	0.4976	0.9659	Nucleolar
17.4949	8.0685	4.4688	0.0596	0.8505	0.5474	0.9702	Nucleolar
18.0377	8.8217	4.5376	0.0494	0.8827	0.5514	0.9753	Nucleolar
16.5442	9.0994	4.4790	0.0303	0.9199	0.5932	0.9849	Nucleolar
17.9521	11.4094	4.4717	0.0419	0.9031	0.5818	0.9790	Nucleolar
21.0611	11.1020	4.9101	0.0610	0.8749	0.4608	0.9695	Nucleolar
15.3989	8.8858	4.1740	0.0260	0.9126	0.6775	0.9870	Nucleolar
18.1230	9.8181	4.6013	0.0578	0.8609	0.5436	0.9711	Nucleolar
17.3541	9.7119	4.4717	0.0537	0.8556	0.5930	0.9732	Nucleolar
29.4178	7.9733	4.7879	0.0198	0.8826	0.8169	0.9901	Nucleolar
28.5085	7.3642	4.5784	0.0116	0.7757	0.9366	0.9942	Nucleolar
26.6778	5.5818	4.3397	0.0203	0.7361	0.9030	0.9898	Nucleolar
25.0662	6.5231	4.3230	0.0400	0.7384	0.8086	0.9800	Nucleolar
25.4085	4.6940	4.0903	0.0205	0.4222	0.9442	0.9897	Nucleolar
25.9258	6.1649	4.3822	0.0434	0.5404	0.8638	0.9783	Nucleolar
25.5565	6.1070	4.4048	0.0398	0.7063	0.8266	0.9802	Nucleolar
24.9434	5.9223	4.2879	0.0386	0.7340	0.8174	0.9807	Nucleolar
25.9715	6.3911	4.3704	0.0264	0.7051	0.8844	0.9868	Nucleolar
27.9528	4.1038	3.9320	0.0007	-0.0003	0.9986	0.9997	Nucleolar
30.9975	7.1795	4.6214	0.0050	0.2300	0.9886	0.9975	Nucleolar
26.0183	2.2949	3.1962	0.0006	0.3426	0.9984	0.9997	Nucleolar
29.6535	7.0309	4.6321	0.0199	0.7749	0.8925	0.9901	Nucleolar
27.3909	4.7700	4.1887	0.0211	0.6233	0.9231	0.9894	Nucleolar
25.1200	3.6077	3.8254	0.0209	0.6360	0.9220	0.9896	Nucleolar
27.2929	6.9400	4.5357	0.0247	0.5468	0.9212	0.9876	Nucleolar
23.6113	3.2880	3.6874	0.0305	0.6473	0.8836	0.9847	Nucleolar
26.3090	5.5574	4.1337	0.0086	0.4997	0.9743	0.9957	Nucleolar

28.3625	8.1286	4.8306	0.0226	0.7826	0.8737	0.9887	Nucleolar
24.5380	4.8013	4.0956	0.0420	0.6296	0.8460	0.9790	Nucleolar
25.5362	5.1130	4.1296	0.0226	0.6701	0.9091	0.9887	Nucleolar
21.2766	3.6239	3.7282	0.0660	0.7798	0.6386	0.9670	Nucleolar
22.8020	2.9071	3.4540	0.0237	0.5687	0.9217	0.9881	Nucleolar
24.5215	4.2079	4.0152	0.0447	0.5756	0.8517	0.9776	Nucleolar
22.7689	3.8893	3.8017	0.0326	0.8015	0.8040	0.9837	Nucleolar
24.3112	3.3493	3.6985	0.0135	0.5847	0.9542	0.9933	Nucleolar
19.7600	4.4867	3.9486	0.1076	0.7845	0.4045	0.9462	Nucleolar
22.2000	4.6903	4.0970	0.0602	0.7862	0.6619	0.9699	Nucleolar
20.7381	3.9657	3.7704	0.1225	0.6939	0.4924	0.9388	Nucleolar
19.7437	3.8091	3.6806	0.1090	0.7767	0.4149	0.9455	Nucleolar
24.5583	5.7160	4.2609	0.0552	0.6964	0.7657	0.9724	Nucleolar
25.2459	6.0020	4.3403	0.0392	0.6904	0.8354	0.9804	Nucleolar
24.5437	4.6019	4.0455	0.0366	0.6504	0.8599	0.9817	Nucleolar
27.4221	7.2883	4.3337	0.0129	0.5054	0.9610	0.9935	Nucleolar
24.5888	4.6178	3.9916	0.0279	0.5138	0.9153	0.9860	Nucleolar
26.9507	6.7227	4.2585	0.0110	0.4841	0.9679	0.9945	Nucleolar
25.2231	5.8684	4.2808	0.0316	0.7270	0.8537	0.9842	Nucleolar
25.3913	5.6909	4.2356	0.0305	0.6190	0.8904	0.9848	Nucleolar
26.3841	5.3543	4.1160	0.0100	0.4485	0.9718	0.9950	Nucleolar
26.7787	5.6345	4.2121	0.0141	0.5075	0.9572	0.9929	Nucleolar
25.2563	5.1383	4.0689	0.0172	0.3560	0.9563	0.9914	Nucleolar
24.5836	4.8097	4.0207	0.0326	0.5300	0.8989	0.9837	Nucleolar
27.4867	6.1259	4.0936	0.0022	0.0561	0.9954	0.9989	Nucleolar
25.9492	6.0236	4.5075	0.0233	0.8674	0.8014	0.9884	Nucleolar
23.6759	4.0623	3.8882	0.0438	0.6717	0.8245	0.9781	Nucleolar
23.4451	4.9474	3.9064	0.0599	0.5367	0.8141	0.9700	Nucleolar
24.0233	3.8419	3.8424	0.0282	0.6011	0.9016	0.9859	Nucleolar
25.1534	4.3342	3.9601	0.0126	0.5115	0.9618	0.9937	Nucleolar

APPENDIX D: TESTING SET AND CLASSIFICATION RESULTS

Mean	Std Dev	Entropy	Contrast	Correlation	Energy	Homogeneity	Pattern	Classify
26.9609	6.9971	4.7499	0.0313	0.8472	0.7639	0.9843	Homogeneous	Y
28.0568	10.7516	4.8497	0.0579	0.6911	0.7577	0.9711	Homogeneous	Y
30.2021	7.5778	4.5551	0.0353	0.8184	0.7713	0.9824	Homogeneous	Y
28.0879	6.7043	4.4952	0.0274	0.8759	0.7520	0.9863	Homogeneous	Y
34.8981	9.6658	4.5322	0.0436	0.7758	0.7633	0.9782	Homogeneous	Y
32.0473	9.1229	4.8291	0.0352	0.8385	0.7475	0.9824	Homogeneous	Y
31.1596	6.4054	4.6172	0.0231	0.6745	0.9059	0.9884	Homogeneous	Y
35.2150	9.2133	5.0337	0.0342	0.7762	0.8182	0.9829	Homogeneous	N
36.5095	9.6343	4.7357	0.0343	0.7747	0.8140	0.9828	Homogeneous	Y
35.8668	8.0971	4.6768	0.0270	0.5241	0.9169	0.9865	Homogeneous	N
35.1700	9.6645	5.0740	0.0386	0.5575	0.8766	0.9807	Homogeneous	N
33.4663	8.0523	4.8422	0.0240	0.4784	0.9305	0.9880	Homogeneous	N
33.3923	7.4052	4.6137	0.0211	0.5152	0.9357	0.9895	Homogeneous	N
35.9545	9.2899	4.9536	0.0249	0.6230	0.9098	0.9875	Homogeneous	Y
37.7406	11.2802	5.0239	0.0338	0.8028	0.8064	0.9845	Homogeneous	N
32.7673	10.2915	4.9427	0.0589	0.7544	0.7056	0.9705	Homogeneous	Y
31.5607	8.3287	4.8324	0.0286	0.8539	0.7757	0.9857	Homogeneous	Y
32.7024	9.0208	4.6920	0.0478	0.7334	0.7745	0.9761	Homogeneous	Y
30.2271	7.2115	4.3918	0.0457	0.6409	0.8291	0.9772	Homogeneous	Y
34.7890	9.1683	4.9307	0.0228	0.7053	0.9001	0.9886	Homogeneous	Y
33.8905	8.1453	4.6901	0.0428	0.6277	0.8437	0.9786	Homogeneous	Y
30.5471	5.9085	4.3647	0.0325	0.5992	0.8869	0.9837	Homogeneous	Y
30.5963	7.4608	4.8292	0.0260	0.7148	0.8833	0.9870	Homogeneous	Y
32.2344	8.7689	4.4396	0.0294	0.8852	0.7148	0.9853	Homogeneous	Y
29.1948	7.6225	4.7900	0.0335	0.7413	0.8427	0.9833	Homogeneous	Y
37.4607	10.4798	5.2313	0.0261	0.8088	0.8398	0.9869	Fine Speckled	Y
34.4353	12.2739	5.3005	0.0389	0.8686	0.6716	0.9805	Fine Speckled	Y
31.9345	10.4993	5.2034	0.0361	0.8430	0.7347	0.9819	Fine Speckled	Y
32.5209	10.1689	5.1728	0.0357	0.8380	0.7450	0.9822	Fine Speckled	Y
31.5749	10.5548	5.1973	0.0336	0.8698	0.7087	0.9832	Fine Speckled	Y
31.9083	10.1479	5.2256	0.0303	0.8739	0.7307	0.9849	Fine Speckled	Y
36.8327	9.6095	5.0674	0.0204	0.7568	0.8964	0.9898	Fine Speckled	N
37.8653	11.4714	5.2155	0.0406	0.7662	0.7936	0.9797	Fine Speckled	Y
33.0907	12.4651	5.1976	0.0324	0.9027	0.6359	0.9838	Fine Speckled	Y
36.2631	12.4819	5.2167	0.0435	0.8161	0.7280	0.9782	Fine Speckled	Y
36.1678	10.6848	5.0725	0.0353	0.8084	0.7815	0.9823	Fine Speckled	Y
39.9279	10.9201	5.2686	0.0314	0.7982	0.8141	0.9843	Fine Speckled	Y
38.8951	11.9910	5.2296	0.0511	0.7664	0.7498	0.9745	Fine Speckled	Y

34.6487	11.8116	5.2913	0.0318	0.8807	0.7122	0.9841	Fine Speckled	Y
37.7545	11.1302	5.3511	0.0351	0.8109	0.7941	0.9825	Fine Speckled	Y
37.3272	10.2719	5.0810	0.0066	0.6304	0.9755	0.9967	Fine Speckled	N
37.2334	8.4922	4.9623	0.0052	0.3916	0.9862	0.9974	Fine Speckled	Y
35.8908	10.1235	5.0089	0.0185	0.4599	0.9479	0.9908	Fine Speckled	Y
38.4903	8.9061	5.0114	0.0053	0.5390	0.9833	0.9974	Fine Speckled	N
38.9800	11.7818	5.3373	0.0499	0.7302	0.7808	0.9751	Fine Speckled	Y
36.8344	8.6173	4.8006	0.0010	0.3972	0.9973	0.9995	Fine Speckled	Y
35.6502	9.0466	4.9193	0.0026	0.3420	0.9935	0.9987	Fine Speckled	Y
34.8613	9.8202	5.1225	0.0088	0.9377	0.8499	0.9956	Fine Speckled	Y
32.5806	7.4976	4.6864	0.0175	0.5800	0.9409	0.9912	Fine Speckled	N
30.3167	9.4759	4.9391	0.0382	0.8254	0.7442	0.9809	Fine Speckled	N
42.3863	16.7237	5.7428	0.0964	0.7845	0.4966	0.9518	Coarse Speckled	Y
41.5145	15.9052	5.7179	0.1210	0.7573	0.4686	0.9395	Coarse Speckled	Y
39.6827	14.2441	5.5914	0.1104	0.7250	0.5601	0.9448	Coarse Speckled	Y
40.6521	12.4755	5.5744	0.0790	0.7323	0.6504	0.9605	Coarse Speckled	Y
39.7142	12.6715	5.5340	0.0823	0.6966	0.6709	0.9589	Coarse Speckled	Y
37.9712	11.2372	5.3942	0.0591	0.7294	0.7408	0.9704	Coarse Speckled	N
34.4986	10.5829	5.3166	0.0502	0.6392	0.8287	0.9764	Coarse Speckled	N
45.5115	16.6057	5.8400	0.0985	0.7962	0.4614	0.9526	Coarse Speckled	Y
40.6701	13.3809	5.5729	0.0758	0.7254	0.6563	0.9621	Coarse Speckled	Y
40.0386	14.5478	5.6368	0.1133	0.7143	0.5481	0.9434	Coarse Speckled	Y
40.6414	12.4549	5.4530	0.0697	0.7219	0.6893	0.9651	Coarse Speckled	Y
40.7599	15.3956	5.6371	0.1173	0.7829	0.4628	0.9413	Coarse Speckled	Y
37.2749	13.6568	5.5637	0.0944	0.7669	0.5770	0.9528	Coarse Speckled	Y
36.8380	12.3907	5.4283	0.0889	0.7231	0.6389	0.9556	Coarse Speckled	Y
45.0079	17.4574	5.8396	0.1046	0.7733	0.4670	0.9477	Coarse Speckled	Y
46.6312	14.6493	5.7540	0.1085	0.7407	0.4951	0.9458	Coarse Speckled	Y
39.3975	14.1537	5.5172	0.1168	0.6652	0.5896	0.9416	Coarse Speckled	Y
46.3719	15.7178	5.8141	0.1216	0.7660	0.4117	0.9392	Coarse Speckled	Y
45.8950	18.9748	5.9270	0.1117	0.8193	0.3956	0.9442	Coarse Speckled	Y
41.0990	14.8330	5.6508	0.1172	0.7402	0.5132	0.9414	Coarse Speckled	Y
35.8006	9.9177	5.1887	0.0355	0.4741	0.8999	0.9822	Coarse Speckled	N
37.7258	11.7702	5.5095	0.0656	0.6875	0.7479	0.9673	Coarse Speckled	Y
38.3997	14.4203	5.7142	0.0859	0.7705	0.5988	0.9570	Coarse Speckled	Y
55.7739	20.1523	6.2600	0.1048	0.8447	0.3809	0.9476	Coarse Speckled	Y

39.5414	11.0690	5.3648	0.0315	0.8194	0.7964	0.9842	Coarse Speckled	N
19.4674	14.5773	4.6153	0.0928	0.8587	0.5241	0.9554	Centromere	Y
17.7515	9.1569	4.4975	0.0927	0.7944	0.5137	0.9537	Centromere	Y
16.6563	8.2945	4.4161	0.0799	0.8038	0.5374	0.9601	Centromere	Y
15.0066	9.2469	4.3649	0.0604	0.8270	0.6438	0.9698	Centromere	Y
15.6100	9.3804	4.3879	0.0693	0.8185	0.6051	0.9654	Centromere	Y
31.4670	15.2497	5.7515	0.0782	0.8504	0.5176	0.9609	Centromere	N
16.2374	8.5113	4.3691	0.0794	0.7963	0.5715	0.9603	Centromere	Y
16.6214	8.6531	4.3980	0.0893	0.7798	0.5551	0.9554	Centromere	Y
20.0294	10.0584	4.8111	0.1018	0.8123	0.4278	0.9491	Centromere	Y
18.8585	8.9067	4.7429	0.0889	0.8179	0.4525	0.9555	Centromere	Y
20.1515	12.2947	4.8405	0.0818	0.8698	0.4260	0.9592	Centromere	Y
22.4177	11.8513	5.0483	0.0825	0.8625	0.4099	0.9587	Centromere	Y
21.3409	12.7671	5.1066	0.0685	0.8913	0.4291	0.9658	Centromere	Y
22.2354	13.2041	5.0641	0.0793	0.8787	0.4174	0.9604	Centromere	Y
26.1014	13.5089	5.3202	0.0975	0.8360	0.4305	0.9512	Centromere	Y
20.9308	12.9022	4.9265	0.0979	0.8546	0.4118	0.9516	Centromere	Y
20.7596	9.1596	4.7579	0.0771	0.8494	0.4329	0.9615	Centromere	Y
20.6524	10.6886	4.5154	0.1013	0.8207	0.4445	0.9494	Centromere	Y
20.4744	12.4753	4.9103	0.1011	0.8346	0.4280	0.9496	Centromere	Y
17.4677	10.6033	4.7155	0.0923	0.8057	0.5175	0.9538	Centromere	Y
19.1540	10.9722	4.7249	0.0778	0.8542	0.4754	0.9611	Centromere	Y
19.5595	13.3442	4.8328	0.0908	0.8547	0.4708	0.9553	Centromere	Y
19.3050	8.3335	4.6886	0.0989	0.8072	0.4211	0.9505	Centromere	Y
21.4162	10.1285	4.9887	0.0967	0.8221	0.4073	0.9516	Centromere	Y
22.0359	14.0392	5.1394	0.1033	0.8530	0.3846	0.9484	Centromere	Y
22.0523	3.3643	3.5569	0.0719	0.5684	0.7665	0.9641	Nucleolar	Y
23.1987	4.0471	3.8741	0.0433	0.7065	0.8110	0.9784	Nucleolar	Y
21.2010	3.4848	3.6879	0.0858	0.7482	0.5805	0.9571	Nucleolar	Y
23.0764	3.7731	3.6243	0.0357	0.6929	0.8492	0.9821	Nucleolar	Y
23.5704	3.0538	3.3604	0.0108	0.4843	0.9684	0.9946	Nucleolar	Y
22.4192	3.0415	3.4071	0.0401	0.5576	0.8706	0.9799	Nucleolar	Y
22.0565	2.9541	3.4044	0.0446	0.6723	0.8210	0.9777	Nucleolar	Y
21.1564	3.9914	3.5756	0.0624	0.7976	0.6330	0.9688	Nucleolar	Y
21.6529	3.4783	3.5647	0.0649	0.7311	0.6972	0.9675	Nucleolar	Y
20.0243	4.3435	3.8994	0.0902	0.8090	0.4455	0.9549	Nucleolar	N
21.9630	5.3213	4.0989	0.0785	0.7898	0.5540	0.9608	Nucleolar	N
22.5429	5.1716	3.9584	0.0643	0.7634	0.6681	0.9679	Nucleolar	Y
22.5181	4.5058	3.9044	0.0661	0.7187	0.7032	0.9670	Nucleolar	Y
21.2816	4.1507	3.8593	0.0978	0.7311	0.5479	0.9511	Nucleolar	N
22.9899	5.6602	4.1187	0.0677	0.7488	0.6669	0.9661	Nucleolar	Y

24.6969	5.5091	4.2334	0.0387	0.7299	0.8192	0.9806	Nucleolar	Y
25.2523	5.7844	4.2676	0.0379	0.6743	0.8468	0.9810	Nucleolar	Y
23.1726	2.9270	3.5039	0.0371	0.7083	0.8368	0.9814	Nucleolar	Y
23.0056	4.5204	3.9364	0.0347	0.8210	0.7727	0.9827	Nucleolar	Y
22.2283	4.4128	3.9187	0.0829	0.7196	0.6283	0.9586	Nucleolar	N
21.8490	3.8752	3.3729	0.0790	0.5441	0.7539	0.9605	Nucleolar	Y
22.0326	3.4046	3.5698	0.0418	0.7680	0.7792	0.9791	Nucleolar	Y
23.3954	2.7962	3.3329	0.0085	0.4815	0.9752	0.9958	Nucleolar	Y
21.5674	2.6706	3.3162	0.0673	0.5867	0.7743	0.9664	Nucleolar	Y
22.2710	4.7803	3.9262	0.0626	0.7700	0.6688	0.9687	Nucleolar	Y

APPENDIX E: CODING FOR GUI

```
function varargout = GUI(varargin)
% GUI MATLAB code for GUI.fig
%      GUI, by itself, creates a new GUI or raises the existing
%      singleton*.
%
%      H = GUI returns the handle to a new GUI or the handle to
%      the existing singleton*.
%
%      GUI('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in GUI.M with the given input arguments.
%
%      GUI('Property','Value',...) creates a new GUI or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before GUI_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to GUI_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help GUI

% Last Modified by GUIDE v2.5 01-Dec-2013 07:13:15

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @GUI_OpeningFcn, ...
                   'gui_OutputFcn',  @GUI_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before GUI is made visible.
function GUI_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to GUI (see VARARGIN)

% Choose default command line output for GUI
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes GUI wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = GUI_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
```

```
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

features = [];
[fn pn] = uigetfile('*.png','select png file');
complete = strcat(pn,fn);
image = imread(complete);
ImageGray = rgb2gray(image);
%figure;
%subplot(2,1,1);
%imshow(image), title('Original Image');
%subplot(2,1,2);
%imshow(ImageGray , []); title('Grey Image');
axes(handles.axes1);
imshow(ImageGray);
axes(handles.axes2);
imshow(image);
%% First-order statistics


A=double(ImageGray);

Mean = mean(ImageGray(:));
Variance = var(A(:));
StdDev = sqrt(Variance(:));
Entropy = entropy(ImageGray);


%% GLCM

offsets1 = [0 1;-1 1;-1 0;-1 -1];
glcm = graycomatrix (ImageGray, 'GrayLimits', [0 255], 'NumLevels', 8, 'Offset',
offsets1, 'Symmetric', true);
stats = graycoprops (glcm, 'contrast');
stats2 = graycoprops (glcm, 'correlation');
stats3 = graycoprops (glcm, 'energy');
stats4 = graycoprops (glcm, 'homogeneity');

t = struct2array(stats);
t2 = struct2array(stats2);
t3 = struct2array(stats3);
t4 = struct2array(stats4);

Contrast = mean(t);
Correlation = mean(t2);
Energy = mean(t3);
Homogeneity = mean(t4);


%% Saving Data

features = [features; Mean StdDev Entropy Contrast Correlation Energy Homogeneity];

set(handles.uitable1, 'Data', features);

fismat = readfis('Classification');
out = evalfis([Mean StdDev Entropy Contrast Correlation Energy Homogeneity],fismat);
fprintf('Output = %0.3f \n',out);

    if out<0.201
```

```
        disp = ('Staining Pattern : Homogeneous');
    elseif out<0.401
        disp = ('Staining Pattern : Fine Speckled');
    elseif out<0.601
        disp = ('Staining Pattern : Coarse Speckled');
    elseif out<0.801
        disp = ('Staining Pattern : Centromere');
    else
        disp = ('Staining Pattern : Nucleolar');
    end

set(handles.txt,'String',disp);
% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns popupmenu1 contents as cell
array
%        contents{get(hObject,'Value')} returns selected item from popupmenu1


% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to popupmenu1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



function txt_Callback(hObject, eventdata, handles)
% hObject    handle to txt (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of txt as text
%        str2double(get(hObject,'String')) returns contents of txt as a double


% --- Executes during object creation, after setting all properties.
function txt_CreateFcn(hObject, eventdata, handles)
% hObject    handle to txt (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```