ANALYSIS OF FAILURE IN OFFLINE ENGLISH ALPHABET RECOGNITION WITH DATA MINING APPROACH

RUTHRAKUMAR MUNNIAN

UNIVERSITI SAINS MALAYSIA

2019

ANALYSIS OF FAILURE IN OFFLINE ENGLISH ALPHABET RECOGNITION WITH DATA MINING APPROACH

By:

RUTHRAKUMAR MUNNIAN

(Matrix No.: 128969)

Supervisor:

Dr. Loh Wei Ping

June 2019

This dissertation is submitted to

Universiti Sains Malaysia

As partial fulfillment of the requirement to graduate with honors degree in

BACHELOR OF ENGINEERING (MECHANICAL ENGINEERING)

School of Mechanical Engineering

Engineering Campus

Universiti Sains Malaysia

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Signed…… (RUTHRAKUMAR MUNNIAN)

Date……

Statement 1

This journal is the result of my own investigation, except where otherwise stated. Other sources are acknowledged by giving explicit references. Bibliography/ references are appended.

Signed…… (RUTHRAKUMAR MUNNIAN)

Date……

Statement 2

I hereby give consent for my journal, if accepted, to be available for photocopying and for interlibrary loan, and for the title and summary to be made available outside organizations.

Signed…… (RUTHRAKUMAR MUNNIAN)

Date……

ACKNOWLEDGEMENT

First of all, I would like to express our gratitude to Almighty God for giving us the opportunity and help us endlessly in finishing the Final Year Project. I also would like to offer my dedication to the School of Mechanical Engineering, Universit Sains Malaysia for implementing the Final Year Project course in our curriculum.

I wish to express my sincerest gratitude to my supervisor, Dr. Loh Wei Ping who has guided me throughout the thesis with patience. All her knowledge and guidance were highly useful that made to complete this thesis successfully. Apart from that, she always offers her motivation especially in the time of stagnant progress throughout the Final Year Project process. Without her help, the quality of the solution and steps proposed would not be as high as presently proposed.

Lastly, I am also grateful to have my course mates and seniors who have been supporting me throughout this venture no matter in terms of academic wise or mental support. We always share opinions on the journey to discover more knowledge and information.

TABLE OF CONTENTS

DECLARATION ..... **i**

ACKNOWLEDGEMENT ..... **ii**

TABLE OF CONTENTS ..... **iii**

LIST OF TABLES ..... **vi**

LIST OF FIGURES ..... **vii**

LIST OF ABBREVIATIONS ..... **ix**

ABSTRAK ..... **x**

ABSTRACT ..... **xii**

CHAPTER 1 INTRODUCTION **1**

1.1 Overview 1

1.2 Background 1

1.3 Problem Statement 3

1.4 Objective 4

1.5 Scope of Project 4

1.6 Thesis Outline 5

CHAPTER 2 LITERATURE REVIEW **6**

2.1 Overview 6

2.2 Handwriting Recognition Analysis 6

2.3 Enhancing Handwriting Recognition Accuracy 8

2.4 Issues and Challenges 12

CHAPTER 3 METHODOLOGY **17**

3.1 Overview 17

3.2 Data Collection 18

3.3 Data Transformation 19

3.3.1 Scanning 19

3.3.2 Noise Removal 20
3.3.3 Feature extraction 21
3.3.4 Featured Data Transformation 25
3.4 Data Pre-processing 25
3.5 Data Classification 27
3.6 Classification Error Analysis 28
3.6.1 Phase one 28
3.6.2 Phase two 31
CHAPTER 4 RESULTS AND DISCUSSION 35
4.1 Overview 35
4.2 Data Transformation 35
4.3 Data Pre-Processing 39
4.4 Data Classification 42
4.5 Classification Error Analysis 46
4.5.1 Phase One 46
4.5.1(a) Stroke 47
4.5.1(b) Curve 50
4.5.1(c) Stroke and Curve features 53
4.5.2 Phase Two 62
4.5.2(a) Sharp Vertex 62
4.5.2(b) Closed Region 63
4.5.2(c) Points 64
4.6 Overall Analysis 69
CHAPTER 5 CONCLUSION 72
5.1 Concluding Remarks 72
5.2 Study Contribution 73
5.3 Future Work 74
REFERENCES 75
APPENDIX A: SURVEY LETTER

APPENDIX B: CONCERN FORM

APPENDIX C: SURVEY FORM

APPENDIX D: DATASET FOR SAMPLE 001

LIST OF TABLES

Page

Table 2.1: Summary of researches related to handwriting recognition 12
Table 3.1: Raw dataset layout 25
Table 3.2: Number of stroke and curve for 26 alphabets. 30
Table 3.3: Number of sharp points forming six capital alphabets. 31
Table 3.4: Number of closed regions by alphabets. 32
Table 3.5: Number of points by capital alphabet. 34
Table 4.1: Number of alphabets in each phrase and survey form. 36
Table 4.2: Number of removed alphabets respective to samples showing outliers ... 40
Table 4.3: Number of the alphabet before and after pre-processing with the percentage difference of alphabet after pre-processing 41
Table 4.4: The percentage classification accuracies for before and after preprocessing..... 42
Table 4.5: Classification results for stroke dataset..... 48
Table 4.6: Classification results with added curve attribute 50
Table 4.7: Top 5 best-performed classifiers for the raw and added stroke and curve features to the original dataset..... 53
Table 4.8: Result of the consecutive classifier for all three datasets 54
Table 4.9: Classification error with the wrongly classified alphabet at phase one 58
Table 4.10: Trouble groups of respective alphabets 61
Table 4.11: Classification error with the wrongly classified alphabet at phase two..... 67
Table 4.12: Overall classification accuracy results for the before and after inclusion of attributes..... 69

LIST OF FIGURES

Page

Figure 2.1: Fishbone diagram of handwriting recognition approaches by language and characters 16
Figure 3.1: Project flow chart 18
Figure 3.2: Sample handwritten characters in the survey form 19
Figure 3.3: Epson L220 Scanner 20
Figure 3.4: Survey form (a) before and (b) after noise removal 21
Figure 3.5: Complete MATLAB coding 22
Figure 3.6: The transformation of the raw image 23
Figure 3.7: Bounding box of an alphabet 23
Figure 3.8: Numeric labeling for respective alphabet 24
Figure 3.9: Sample format of the Matlab coding outcomes 24
Figure 3.10: Boxplot with an interquartile range 26
Figure 3.11: Remove with value filter 27
Figure 3.12: Sample stroke count in capital Letter A 29
Figure 3.13: Capital letter B 29
Figure 3.14: Capital letter M 31
Figure 3.15: Capital letter Q 32
Figure 3.16: Capital Letter E 32
Figure 3.17: Number of points in a line 33
Figure 3.18: Number of points in an arc 33
Figure 3.19: Capital letter P with marked with the number of points 33
Figure 4.1: Extracted handwritten data for Sample 001 using MATLAB 38
Figure 4.2: Number of outliers 39
Figure 4.3: Removed outliers of Sample 009 40
Figure 4.4: Relative percentage classification accuracy difference on all algorithms of seven classifiers (After-Before preprocessing) 45
Figure 4.5: Top 5 highest accuracy classifier classification accuracies on all algorithm of seven classifiers 46

Figure 4.6: Relative percentage classification accuracy difference of top 5 algorithms....49
Figure 4.7: The top 5 highest accuracies classifier on added stroke attribute....49
Figure 4.8: Top 5 highest accuracy chart for added curve feature....52
Figure 4.9: Relative percentage difference of top 5 classifiers....52
Figure 4.10: Relative percentage classification accuracy difference comparing the raw dataset with the added stroke(S) attribute and added stroke+curve(SC) attribute.....54
Figure 4.11: Confusion Matrix of IBk Classifier at phase one....57
Figure 4.12: Percentage classification error by the class of alphabets at phase one....58
Figure 4.13: Classification accuracy for the before and after sharp vertex attribute inclusion....63
Figure 4.14: Chart of accuracy for closed region attribute....64
Figure 4.15: Percentage classification accuracy for before and after points attribute inclusion....65
Figure 4.16: Confusion matrix of IBk classifier at phase two....66
Figure 4.17: Classification error chart for the respective alphabet at phase two....67
Figure 4.18: Overall increment chart....70
Figure 4.19: Classification error of alphabet for phase 1 and 2....71

LIST OF ABBREVIATIONS

NNLM	Neural Network Language Model
HMM	Hidden Markov Model
SVM	Support Vector Machine
WEKA	Waikato Environment for Knowledge Analysis
OCR	Optical Character Recognition
MATLAB	Matrix Laboratory
BLSTM	Bidirectional Long Short-Term Memory
R-HOG	Rectangle Histogram Oriented Gradient
FFAN	Feed-Forward Artificial Neural Network
JPEG	Joint Photographic Experts Group
S	Stroke
SC	Stroke + Curve
SCSV	SC + Sharp Vertex
SCSVCR	SCSV + Closed Region
SCSVCRP	SCSVCR + Points

ABSTRAK

Pengenalan tulisan tangan tanpa rangkaian maklumat adalah cara yang digunakan untuk mengenal pasti frasa, huruf atau nombor tulisan tangan sejak zaman dahulu. Kebanyakan kajian sebelum ini dalam bidang pengenalan tulisan tangan ini memberi tumpuan kepada mengenal pasti aksara tulisan tangan menggunakan Neural Network Language Model (NNLM), Hidden Markov Model (HMM), dan Support Vector Machine (SVM) dengan teknik segmentasi, kaedah mengubah Hough dan ciri-ciri struktur. Walau bagaimanapun,cara ini melibatkan algoritma yang kompleks dan memerlukan set data yang besar sebagai modal latihan. Oleh itu, kajian ini menganalisis kegagalan dalam pengenalpastian aksara tulisan tangan mengunakan amalan memeriksa pangkalan data yang besar yang sedia ada untuk menghasilkan maklumat baru. Objektif kajian ini adalah untuk menambah baik cara pengenalan aksara untuk mengklasifikasikan huruf bahasa Inggeris dan menentukan punca kegagalan klasifikasi huruf Bahasa Inggeris dalam tulisan tangan. Kajian ini menggunakan data yang ditulis oleh 50 pelajar Universiti Sains Malaysia yang mengandungi huruf besar bahasa Inggeris. Data ini telah diproses terlebih dahulu untuk mengelakkan data yang berbeza dengan ketara daripada data-data yang lain sebelum analisis klasifikasi dengan bantuan alat Waikato Environment for Knowledge Analysis (WEKA). Analisis klasifikasi pada awalnya dilakukan pada semua tujuh algoritma klasifikasi pada mod pengesahan bersilang 10 kali ganda. Pada fasa pertama, garisan dan lengkung telah ditambah ke dalam set data dan dikelaskan masing-masing. Pada fasa kedua, puncak tajam, huruf tertutup, dan butiran ditambah dalam set data. Tiga algoritma klasifikasi tebaik telah dipilih: IBk, LMT dan Random Committee untuk klasifikasi selanjutnya. Keputusan yang telah diklasifikasi kemudiannya dianalisis untuk mengenalpasti punca kesilapan klasifikasi. Keputusan klasifikasi set

data yang asal menunjukkan ketepatan klasifikasi yang rendah iaitu 25%. Selepas semua artibut tambahan ditambah dalam set data masing-masing, ketepatan klasifikasi telah berjaya ditingkatkan menjadi 89%. Secara keseluruhannya, ketepatan klasifikasi bergantung pada atribut tambahan yang ditambah pada set data yang asal untuk membezakan ciri-ciri huruf tulisan tangan.

ABSTRACT

Offline handwriting recognition is a long existing approach to identify the handwritten phrase, letters or digits. Earlier studies in the handwriting recognition field were mostly focused on recognizing characters using Neural Network Language Model (NNLM) classifier, Hidden Markov Model (HMM), and Support Vector Machine (SVM) with segmentation technique, Hough Transform method, and structural features. However, these approaches involve complex algorithms and require voluminous dataset as the training model. Therefore, this study attempts a data mining approach to the analysis of failure in offline English alphabet recognition. The objectives of the study are to improve the pattern recognition approach for classifying English alphabets and to determine the root of classification failure in handwritten English alphabets. Handwritten data of capital letters of the English alphabet by 50 Universiti Sains Malaysia student experimented. The data was pre-processed to remove the outliers prior to classification analysis with the aid of the Waikato Environment for Knowledge Analysis (WEKA) tool. Classification analysis was initially performed on all seven classifier's algorithms at 10-fold dross validation mode. At phase one, Stroke and Curve are added into the dataset and classified respectively. At phase two, Sharp Vertex, Closed Region, and Points are added in the dataset. The top three classification algorithms were selected: IBk, LMT and Random Committee for further classification. The classified result was further analyzed to identify the root of classification errors. At the raw dataset classification, the classification accuracy is low with 25%. As the attributes are added to raw dataset respectively, the accuracy of classification was successfully increased to 89%. Conclusively, the accuracy of the classification depends on the added attributes to distinguish characteristics of the alphabets.

CHAPTER 1 –INTRODUCTION

1.1 Overview

This project studies the analysis of failure in offline English alphabet recognition with a data mining approach. Handwriting is the process of recording an important note, message or information with writing instruments such as pen or pencil on a piece of paper. Handwriting serves as the fundamental skills that will be used throughout the whole life for learning and recording information. Handwriting can be categorized into two types: offline and online handwriting. However, these studies only focus on offline handwriting. The purpose of this study was to enhance the English character classification analysis and to identify the root of the classification failures in the handwritten English characters. The study involves 50 participants of Universit Sains Malaysia Engineering Campus students on a voluntary basis. The study attributes include sample, age, gender, handedness, the height of the alphabet, width of the alphabet, true area of the alphabet.

1.2 Background

In the digital world, handwriting character is commonly recognized and classified from the image analysis and feature recognition bases. Handwriting recognition is a method used to transform the handwritten data on letters, cheques, diaries or notes into useful computerized data. The process of transforming the data is commonly aided by hardware such as scanner, camera, computer and phone and software such as simple Optical Character Recognition (OCR), Top OCR and OCR desktop. The scanned handwritten documents known as data input will be interpreted and transformed into digitalized data as the output.

As human handwriting varies by individual characters, research works have been extensively conducted in conventional offline handwritings focusing on different techniques applied to recognize and interpret handwriting characters. There are a number of techniques used for the handwriting character recognition such as neural network technique[1, 2], ensemble classifiers[3], combining dissimilar classifier, normalization[4], binarization[5]. The common characters examined were English[5, 6], Malayalam[7, 8], Gujarati[2] and numbers[4]. All the studies were concentrated on the effectiveness of different approaches to distinguish the handwriting patterns in the process to recognize them. In the handwriting recognition process, several attributes considerably affecting the handwriting patterns include the demographic factors like age, gender and handedness and external factors like the roughness of paper and written surface. These conditions were also reported in Bal and Saha[9] using segmentation, baseline recognition and writing pressure detection techniques.

Although the handwriting recognition has been successfully reported in existing works, the handwriting recognition is not fully accurate as it falls within the range of 50% accuracy to 99% accuracy. The range of accuracy reported in the literature was based on subgraph matching and Levenshtein distance technique [7]. Such results lead to some doubts and curiosity on the roots of failure in the handwriting recognition technology.

Therefore, the aims of the study were to identify the pattern recognition approaches for improving the English alphabets classification and to evaluate the roots of classification failures in handwriting analysis. The study involves experimental

handwritings of randomly invited 50 participants of the School of Mechanical Engineering, Universiti Sains Malaysia, Engineering Campus. The participants were required to write the given phrase "THE QUICK BROWN FOX JUMPS OVER LAZY DOG" for three repetitions in the prepared grid boxes survey form. The demographic attributes recorded include the gender, age, and handedness along with English characters quantitative attributes measured from the resultant of handwriting scanned samples. The handwriting image conversion into numeric measurements was performed using Matrix Laboratory (MATLAB) R2015a, followed by the classification analysis into 26 English alphabets as the class attribute supported by Waikato Environment for Knowledge Analysis (WEKA) tool. Failures to correctly classify the handwriting patterns will be examined from the confusion matrix executed from the built-in classifiers of WEKA software.

1.3 Problem Statement

There are various approaches used in the handwriting recognition on the diverse languages which involve numerous non-identical characters. However, these studies were mainly focused on exploring new methods to recognize and interpret handwriting. Although the handwriting character recognition has long existence, the recognition accuracy can be improved. Furthermore, there are no research works conducted on the analysis of failures in handwriting recognition using a data mining technique. Failures to determine the distinctive features of the handwriting character contributes to misclassifications. Hence, a study on the failures in handwriting character recognition using data mining technique is deemed necessary to improve pattern recognition success rate.

1.4 Objective

This research aims

- to improve pattern recognition approach for classifying English alphabets.
- to determine the roots of classification failures in handwritten English alphabets

1.5 Scope of Project

The study involves an experimental case study handwriting involving the participation of 50 students of School of Mechanical Engineering, Universiti Sains Malaysia on a voluntary basis. The gender, age, and handedness were the demographic attributes were recorded along with handwriting characteristics' quantitative measures. This study involves the application of data mining techniques which requires the use of MATLAB software for an initial handwriting image processing followed by the transformation of images into numeric data for handwriting classifications using the WEKA software. The collected data will be pre-processed to smoothen the noisy data as well as to remove the outliers. The data classification approach involves the English characters categorization into classes of 26 alphabets on the identified handwriting patterns considering the capital and small letters.

1.6 Thesis Outline

This thesis is structured into 5 chapters. Chapter 1 presents the introduction of handwriting that begins with the project background. In this chapter, the objectives, study scope, and problem statement are presented.

Chapter 2 discusses the literature review based on the published information from the previous study. The topics that are related to this study is extracted in order to explore the existing studies and the gap of analysis. The issues and challenges encountered in previous studies are considered.

Chapter 3 explains the entire methodology processes involved four levels of processes. The methodology processes involve the data collection, data preprocessing, data classification and data error analysis.

Chapter 4 delivers the results obtained from the offline handwriting analysis beginning from the transformation of collected handwriting samples into numeric forms followed by data preprocessing, classification and classification error analysis.

Lastly, Chapter 5 summarises the overall conclusion from findings obtained in this study. Study contributions and how the objectives of this study are achieved as well as the future directions of the study are discussed.

CHAPTER 2 – LITERATURE REVIEW

2.1 Overview

This chapter focuses on the existing works related to handwriting recognition which covers different languages ranging from English to Arabic, Malayalam, Marathi, Persian, Gujarathi and Chinese handwriting. In previous studies, the researchers were keen on proposing various methods to recognize handwriting. Most research works were focused on neural network classifiers. Apart from that, there are various methods adopted to increase classification accuracies such as segmentation, binarization, and normalization.

2.2 Handwriting Recognition Analysis

Handwriting is a fundamental skill of an individual in life. There are abundant of studies been carried out in the field of offline handwriting recognition on various languages. The offline character recognitions commonly uses optical character recognition (OCR) in the recognition process where the characters that were written on papers are recognized by the OCR software. A various field such as automatic postal readers, office automation and automatic data entry from paper documents to a computer are used the OCR [10]. Offline handwriting recognition system receives data from scanned handwritten documents and applies an image processing technique for analyzing a specific character.

The analysis was reported in various aspects of studies specifically using a different type of classifiers, language, and technique for character recognition,

Researchers commonly used are several classifiers for classifying the handwritten characters. Among the classifiers used include Neural Network classifier, Hidden Markov Model (HMM), Support Vector Machine, Template Matching, and

Bagging classifier. However, Neural Network classifier[1, 2, 5, 6, 8, 11-13] is the most widely used classifiers among the other classifiers such as Hidden Markov model (HMM)[14-17], Support Vector Machine(SVM)[7, 8, 12], Ensemble Classifiers, Subgraph and Levenshtein distance[18] and Template Matching scheme[19, 20].

Various approaches were also integrated into handwriting character recognition. For instance Choudhary et al.[5] uses the binarization technique with neural network classifier for handwritten English character recognition. The authors focused on the training sample quantity, feature extraction technique, and classifier as the main criteria for achieving good accuracy. The technique used resulted in 85.65% accuracy.

Zamora-Martinez et al. [13], whereas, research neural network language model for offline handwriting recognition. The system used two different recognition system: Recurrent Neural Network(NN) and Hybrid HMM/ANN model. Based on their study, the hybrid HMM/ANN system was found capable of dealing with large vocabularies compared to Bidirectional Long Short Term Memory (BLSTM) neural network as it remarks low error rate.

Alex and Das [8] use dissimilar classifier approach for the classification of Malayalam handwritten character. The combination of Neural Network and Support Vector Machine(SVM) resulted in an accuracy of 89.2% for character recognition and 81.2% for handwritten sentence recognition. Their findings showed that the major factor that contributes to accuracy is the method of extracting speeded up robust features, curvature, and diagonal feature.

In a study conducted by Guyon [11], both Neural Network and Naïve Bayes classifiers were used to recognize the handwritten characters. Their study resulted in

Neural Network classifier recorded the lowest training and generalization error compared to Naïve Bayes.

Bal and Saha [9] put emphasis on the use of segmentation, skew recognition and writing pressure detection on the handwriting analysis. Apart from that, Choudhary et al. [21] also dedicated a study on the segmentation approach. The study suggested that the segmentation is good to be used when the character in a word image does not touch each other. As for the Chinese handwriting, the segmentation free strategy with HMM classifier was used to enhance accuracy[22]. Yanikoglu and Sandon [23] also dedicated a study on the segmentation approach. They found a successive segmentation point by evaluating a cost function at each point along the baseline.

Neo et al. [24] dedicated a study to determine the handwriting stroke types and directions for early detection of handwriting difficulties. Basic algorithms were used to identify the types and directions of stroke written according to the input data. The recognition of the writing strokes used as part of an assessment tool to evaluate handwriting performance based on conformity with conventional alphabet writing rules.

2.3 Enhancing Handwriting Recognition Accuracy

There were several techniques to determine accurate handwriting recognition. Among the technique were segmentation, structural feature extraction, R-HOG feature, foreground pixel feature, binarization, vertical projection method, skew recognition, and geometrical feature.

Based on the research works of Yang et al[6], handwriting features combination can produce a better recognition accuracy. The combination of both

structural and statistical feature could enhance the classification of English character on Back Propagation neural network classifier. This approach recorded a perfect 100% accuracy.

Shanjana and James[7] optimized the accuracy of recognition based on the vertical projection method and connected component analysis that uses SVM classifier for the classification of Malayalam handwritten text. Based on handwritten curves word recognition in Dasgupta et al. [25], the Arnold transform is used for scrambling for binary image and Hough transform was used for the stroke orientation feature. Based on their study, large training sample was found to produce a better accuracy as high as 89.6%.

The study in Persian handwritten digit recognition using ensemble classifiers[3] had recommended the usage of ensemble classifiers to boost accuracy. Ensemble classifiers which are bagging classifier and boosting classifiers produce an accuracy of 95.20% for handwritten Persian digit recognition. On the other hand, Putra and Suwardi[18] prove that Levenstein distance classifiers can produce even better accuracy than subgraph classifier for both digit and English character recognition.

Research works of Samanta et al. [17] recommended limiting the vocabulary recognition of unconstrained handwriting based on a Hidden Markov Model (HMM). Their approach produced the highest accuracy due to a large number of parameter and high speed of convergence. Meanwhile, Bertolami and Bunke[14] dedicated to a study on HMM-based on ensemble method for handwritten text line recognition. In the study bagging, random subspace and language model variation method implemented to obtain good recognition accuracy. It is shown that the ensemble method can improve the recognition accuracy over an optimized single reference recognizer.

Another approach that used the combination of HMM and harmony search algorithm has been studied by Zarro and Anwer[15]. Markov model was used as an intermediate classifier and harmony search recognizer used a dominant and common movement of the pattern as a fitness function. This approach records 93.52% of the accuracy of recognition.

Singh et al [20] proposed a new method of classifying handwritten words recognition which uses the template matching scheme. Besides, Choudhury et al. [19] proposed a similar method for handwritten Bangla digit. The study suggested the need for a large training set as the correlation coefficient to achieve high accuracy in handwriting recognition.

While many researchers focused on Neural Network and Hidden Markov Model (HMM), Impedovo et al. [26] also proposed a novel prototype generation technique for handwriting digit recognition. At the first stage, Adaptive Resonance Theory 1 (ART 1) was used for determining the number of prototypes. ART1 is focused on the Evolution Strategy (ES) convergence. Fine tuning the design to generate the best prototype takes place in the second phase.

In research done by Alabodi and Li [27], the study that concerned Arabic handwriting recognition recommended some geometrical features to be used for an effective and efficient with binarization algorithm. Such condition with user parameter could record a high accuracy like in Arabic handwriting recognition by Parvez and Mahmoud[28].

Apart from that, Rabi et al. [16] dedicated a study on recognition of cursive Arabic handwritten text using embedded training based on HMM. The proposed system used HMM with explicit segmentation used embedded training to perform and enhance the character model. The extracted features were based on the densities of the

foreground pixel, concavity, and derivative feature. AlKhateeb and Jinchang [29], whereas, proposed another approach on word recognition based offline recognition system using HMM. In the study, a set of intensity features extracted from each segmented words and structural features of the Arabic characters were extracted including the number of subwords and diacritical marks.

Saadoon and Mohammed [30] proposed a system to identify handwritten Arabic words as one entity without segmentation. In this study, speeded up robust feature (SURF) algorithm were for feature extraction and k-nearest neighbor (KNN) algorithm for classification of the alphabet. The study emphasizes that the success of recognition dependent on various factors such as pre-processing, feature extraction, recognition technique, and classifier being used.

Wang and Tang [4], instead, put emphasis on the normalization process that helps in standardizing the handwriting data. According to their research, the normalization process makes the learning, testing, and recognition easy. Moreover, it is stated that the key success of the approach was the method of handling data from the viewpoint of coordinate instead of image processing.

Desai's [2] suggested improvising the feature abstraction technique and pre-processing technique for better accuracy for Gujarati handwritten digits recognition. With the feature extraction that uses four different profile: horizontal, vertical, and two diagonal, the researcher manages to achieve an accuracy of 81.66% using the Neural Network Classifier. Rectangle histogram oriented gradient (R-HOG) feature was utilized in Kamble and Hegadi[12]. The feature applied both Feed-forward artificial neural network FFANN and SVM for the purpose of comparison of the efficiency of the classifier. FFANN classifier provides a slight higher accurate result of 97.15% than SVM with 95.64% of accuracy.

2.4 Issues and Challenges

Based on the previous research works, it has been learned that most of the researchers used Neural n=Network classifiers for the classification process. Apart from that, the Hidden Markov Model is identified as among the popular classifiers used. The overview of all researchers approaches to handwriting recognition are shown in Table 2.1 and Figure 2.1. The efforts of the researchers on discovering various models for handwriting recognition is essential and helpful provided if the outcome shows reliable accuracy. Existing approaches were wide but there is no generic rule which approaches guarantees good accuracy results in the classification of any handwritten words.

Researchers have also faced a few challenges in the handwriting recognition research area. First of all, there were abundant data required for the training model to assure better recognition. A large amount of data also requires a large amount of time for pre-processing and a large space of storing the data. Apart from that, the complexity of the recognition system or model requires more passions and efforts while building a complete efficient model.

Table 2.1: Summary of researches related to handwriting recognition

Class		Author
Language	Arabic	Abode and Li [27]
		AlKhateeb et al.[29]
		Parvez and Mahmoud[28]
		Rabi et al.[16]
	Bangla	Choudhury et al.[19]
		Samanta et al.[17]
	Chinese	Su et al.[22]
	English	Bal and Saha [9]
		Bertolami and Bunke[14]
		Choudhary et al.[5]
		Dasgupta et al.[25]
		Impedovo et al.[26]
		Martinez et al.[13]

		Neo et al.[24]
		Pradeep et al.[1]
		Putra and Suwardi[18]
		Singh et al.[20]
		Wang and Tang [4]
		Yang et al.[6]
		Yanikoglu and Sandon[23]
	Gujarati	Desai[2]
	Kurdish	Zarro and Anwer[15]
	Malayalam	Alex and Das[8]
		Shanjana and James[7]
	Marathi	Kamble and Hegadi[12]
	Persian	Karimi et al.[3]
Character	Alphabet	Alabodi and Li[27]
		AlKhateeb et al.[29]
		Parvez and Mahmoud[28]
		Rabi et al.[16]
		Samanta et al.[17]
		Su et al.[22]
		Bal and Saha[9]
		Bertolami and Bunke[14]
		Choudhary et al.[5]
		Dasgupta et al.[25]
		Martinez et al.[13]
		Neo et al.[24]
		Pradeep et al.[1]
		Putra and Suwardi[18]
		Singh et al.[20]
		Yang et al.[6]
		Yanikoglu and Sandon[23]
		Zarro and Anwer[15]
		Alex and Das[8]
		Shanjana and James[7]
		Kamble and Hegadi[12]
		Karimi et al.[3]
	Number/Digit	Choudhury et al.[21]
		Guyon[11]
		Impedovo et al.[26]
		Putra and Suwardi[18]
		Wang and Tang[4]
		Desai[2]
Classifier	Bagging Classifier	Karimi et al.[3]
	Boosting Classifier	
	FATF with set medians classifiers	Parvez and Mahmoud[28]
	Harmony Search Algorithm	Zarro and Anwer[15]
	HMM	

		Samanta et al.[17]
		Bertolami and Bunke[14]
		AlKhateeb et al.[29]
		Rabi et al.[16]
		Su et al.[22]
		Martinez et al.[13]
		Zarro and Anwer[15]
	Hybrid HMM/ANN	Martinez et al.[13]
	Levenshtein Distance	Putra and Suwardi[18]
	Neural Network (NN)	Choudhary et al.[5]
		Martinez et al.[13]
		Pradeep et al.[1]
		Alex and Das[8]
		Guyon[11]
		Impedovo et al.[26]
		Desai[2]
	Feed Forward Artificial NN	Kamble and Hegadi[12]
	Back Propagation Neural Network	Yang et al.[6]
	Subgraph	Putra and Suwardi[18]
	Support Vector Machine (SVM)	Dasgupta et al.[25]
		Alex and Das[8]
		Shanjana and James[17]
	Template Matching	Singh et al.[20]
		Choudhury et al.[19]
	WEKA	Wang and Tang[4]
Technique	Connected Component Analysis	Shanjana and James[7]
	Skew Recognition	Bal and Saha[9]
	Vertical Projection Method	Shanjana and James[7]
	Arnold Transform	Dasgupta et al.[25]
	Binarization	Choudhary et al.[5]
	Concavity Feature	Rabi et al.[16]
	Curvature	Alex and Das[8]
	Derivative Feature	Rabi et al.[16]
	Diagonal Feature	Alex and Das[8]
	Foreground Pixel Feature	Rabi et al.[16]
	Geometrical Feature	Alabodi and Li[27]
	Hough Transform	Dasgupta et al.[25]
	Normalization	Wang and Tang[4]
	R-HOG Feature	Kamble and Hegadi[12]
	Segmentation	Yanikoglu and Sandon[23]
		Bal and Saha[9]
		Alex and Das[8]

	Speeded Up Robust Feature (SURF)	Saadoon and Mohammed [30]
	Structural Features	Putra and Suwardi[18]
	Syntactic Pattern Attribute	Parvez and Mahmoud[28]
	Writing Pressure	Bal and Saha[9]

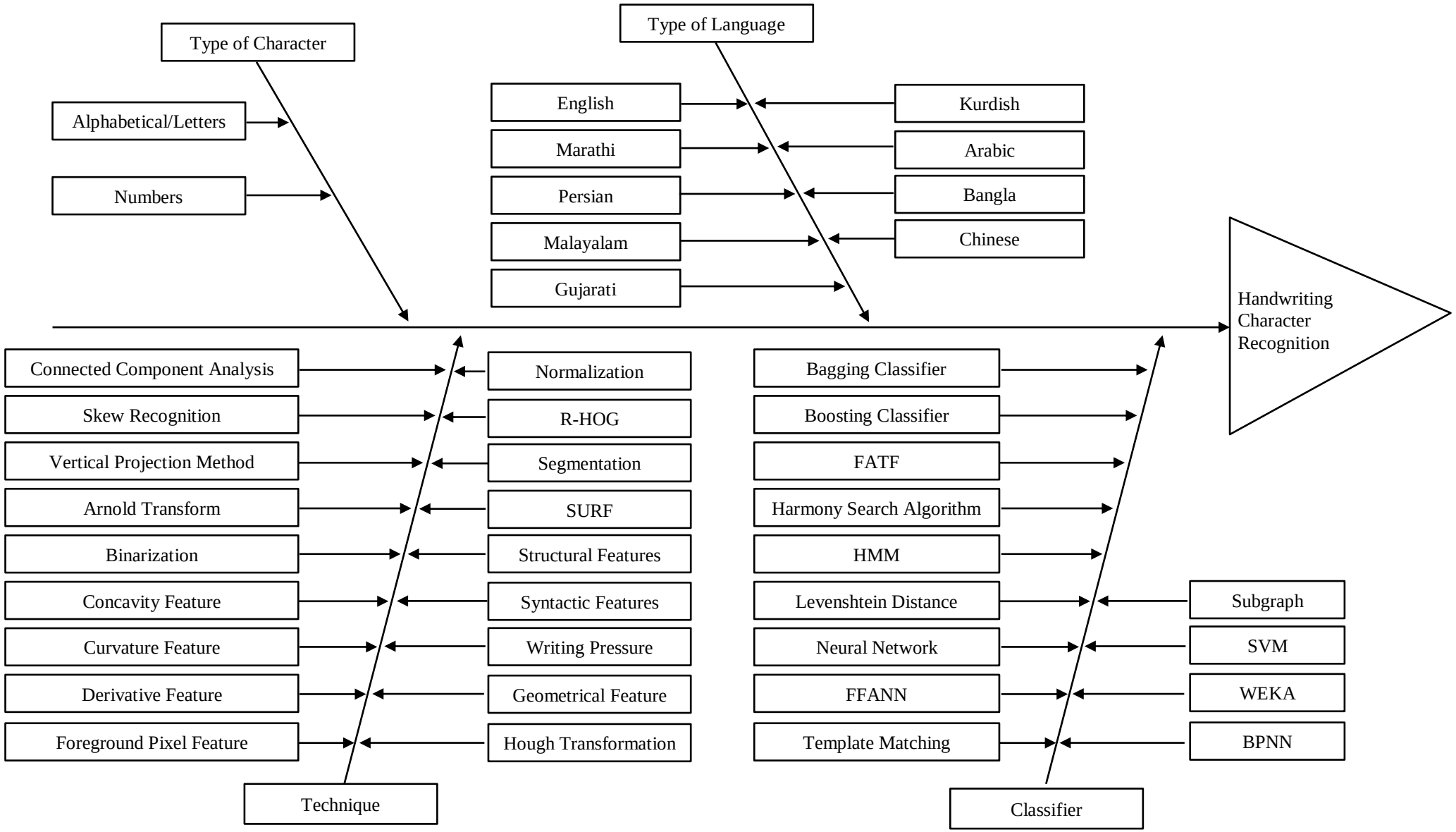
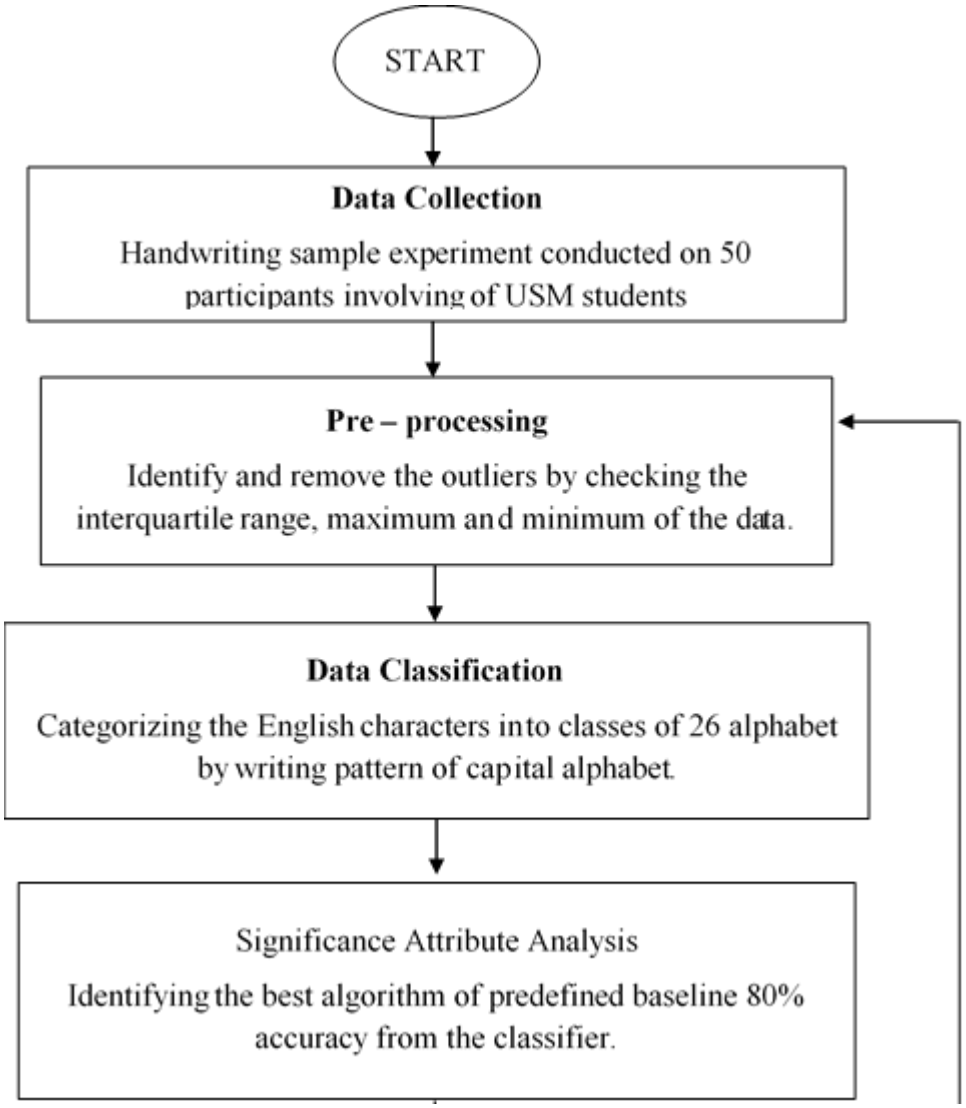


Figure 2.1: Fishbone diagram of handwriting recognition approaches by language and characters

CHAPTER 3 – METHODOLOGY

3.1 Overview

This project applies data mining technique to investigate the failure in handwriting character recognition. The investigation involves four stages of data analysis as shown in Figure 3.1. The research begins with data collection where an experiment was designed to collect the sample handwriting data from 50 Universiti Sains Malaysia students. Collected data were pre-processed for removing noise and outliers. At data processing stage classification analysis was performed on the measurable attributes: height, weight, and size of the letter, slant, and connection of strokes with the algorithms embedded in WEKA tool. The final step was knowledge discovery analysis in order to find the roots of failures for an accurate handwriting character recognition.



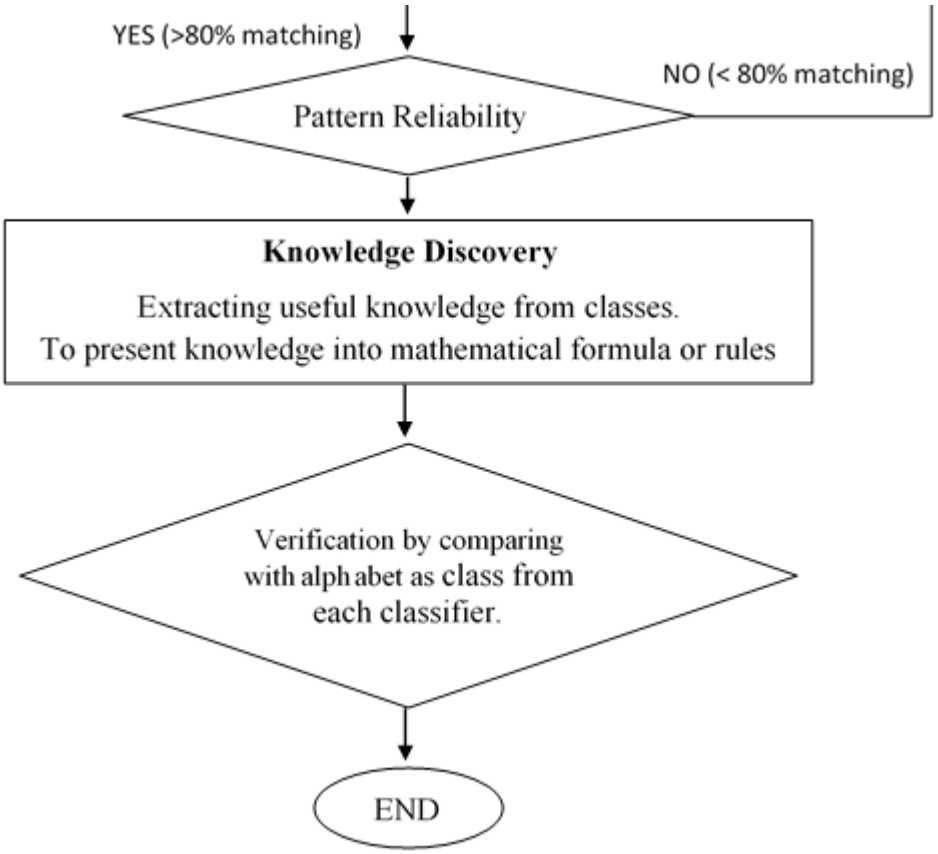


Figure 3.1: Project flow chart

3.2 Data Collection

The case study data consist of handwriting characters involving capital letters of English alphabets (A to Z). The data were collected from 50 participants of USM Engineering Campus students on a voluntary basis. The participants were required to write the phrase "THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG" in the provided grid box column as shown in survey form as shown in Figure 3.2. Other relevant demographics attributes include gender, handedness and age were also recorded.

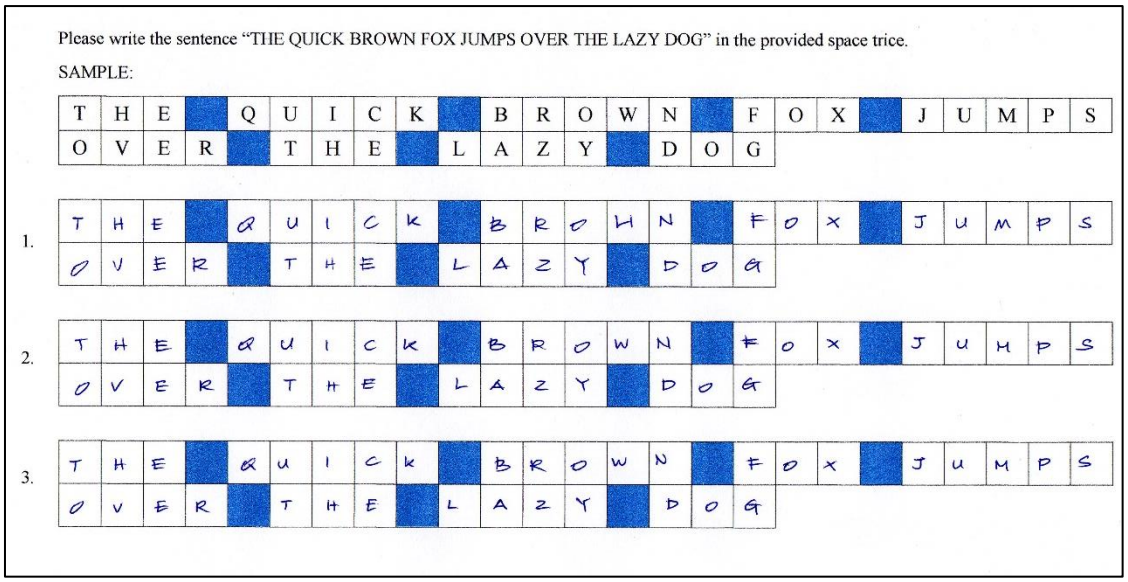


Figure 3.2: Sample handwritten characters in the survey form

3.3 Data Transformation

Based on the collected handwriting samples, several measurable features that include the letter height, width and true area of the letter were extracted. True area refers to is the written alphabet region by the respondent on the survey form. The measurable feature extraction requires four stage processes: handwriting scanning, noise removal with the aid of Adobe Photoshop CS6, characteristic extraction using MATLAB R2015a and the output data transfer into .csv format. The transformed data were subjected to data pre-processing analysis.

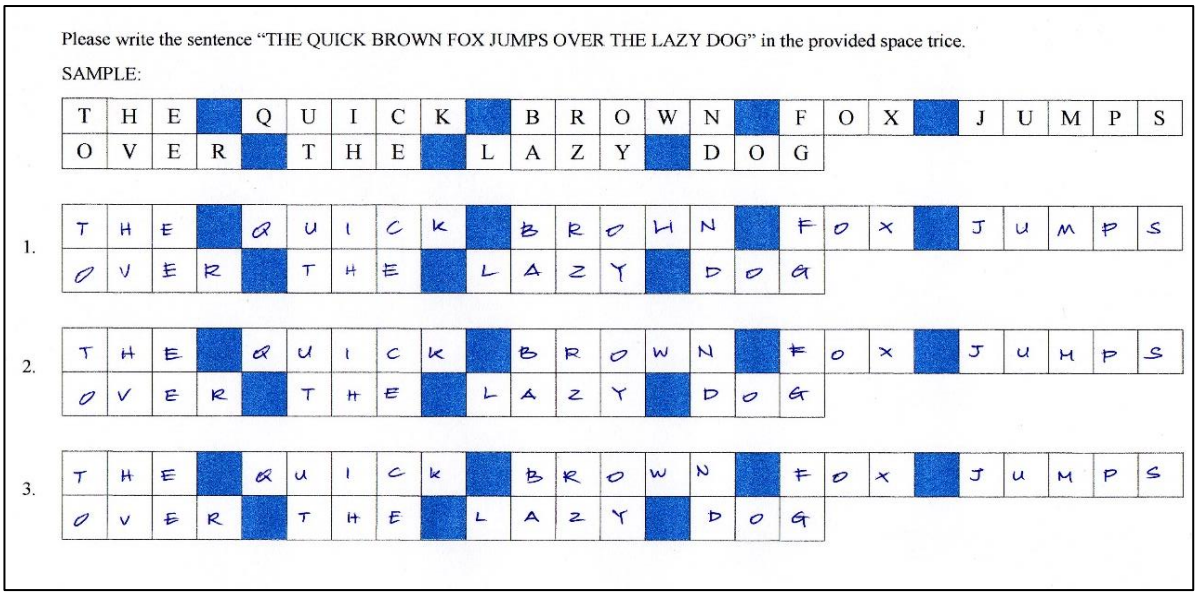
3.3.1 Scanning

The hardcopy survey forms were scanned using EPSON L220 scanner as shown in Figure 3.3. The scanned documents were translated into A4 dimension of Joint Photographic Experts Group (JPEG) format (Figure 3.2),

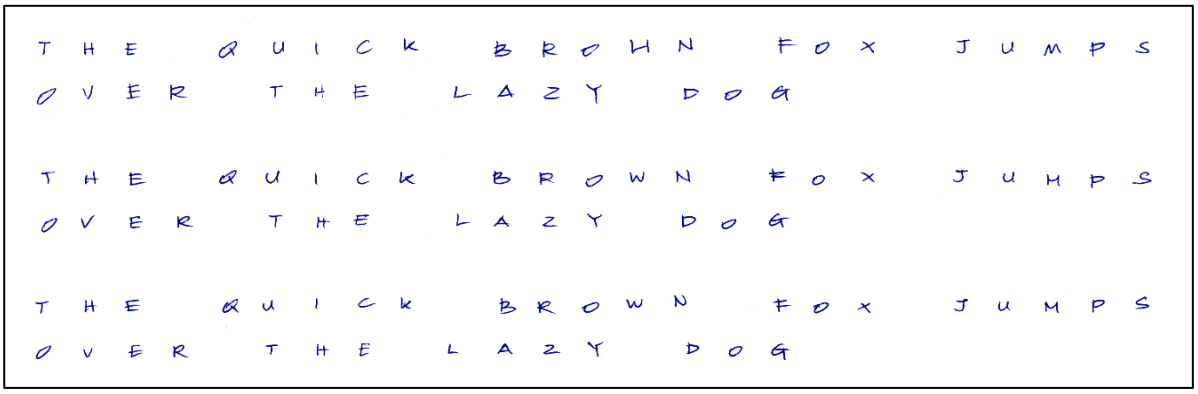
Figure 3.3: Epson L220 Scanner

3.3.2 Noise Removal

As shown in Figure 3.2, the scanned survey form was made up of handwritten characters in grid boxes and border lines which were nondesirable information that potentially interrupt the process of data extraction. Therefore, the lines and the blue boxes on the survey form were removed to ease the process of data extraction. Adobe Photoshop CC 2014 was employed in for removing all the unwanted lines and boxes shown in Figure 3.4(a) and (b).



(a)



(b)

Figure 3.4: Survey form (a) before and (b) after noise removal

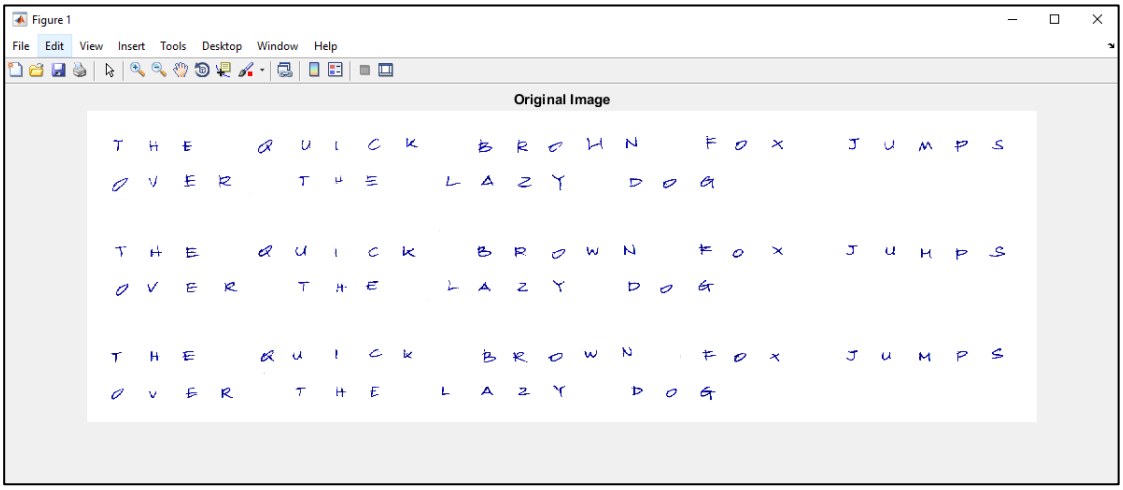
3.3.3 Feature extraction

MATLAB Codings were written for extracting the important data features to characterize patterns of handwriting such as the height, width and true area of the alphabet. The entire MATLAB coding for the handwriting data feature extraction is shown in Figure 3.5.

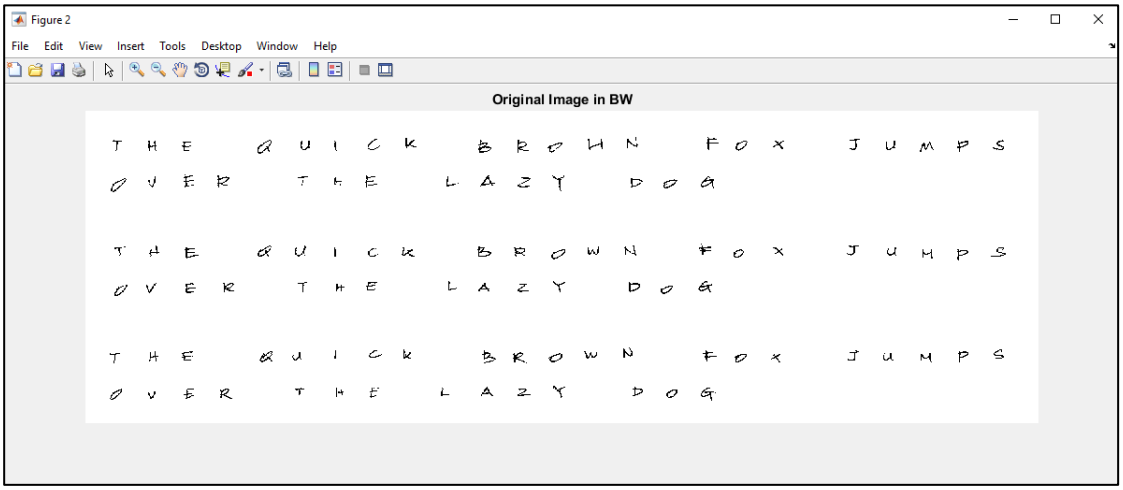
The JPEG image of scanned documents of the survey form after the noise removal process will be imported to the MATLAB. The raw JPEG image will be in the color of the pen (blue or black) used to fill up the form as shown in Figure 3.6(a). Therefore, the RGB image will be converted into grayscale where the hue and saturation information is removed while retaining the luminance and then convert the grayscale image it into the black and white image as shown in Figure 3.6(b). Next is bounding box creation around the alphabet to identify the height, width and true area of the alphabet. The bounding box will be created around the white region assuming the white region is the object. Therefore, the black and white image (Figure 3.6(b)) will be inverse the black and white color as shown in Figure 3.6(c).

```
clc; clear all;
Img1 = imread('Sample001.png');
figure, imshow(Img1), title('Original Image')
I = im2bw(rgb2gray(Img1));
ImgBW = imcomplement(I);
figure,imshow(I),title('Original Image in BW')
[L,num] = bwlabel(ImgBW);
bboxes = regionprops(ImgBW,'BoundingBox');
figure,vislabels(ImgBW), title('Letter Labelling')
figure,imshow(ImgBW),title('Image with Bounding Box')
hold on
stats = regionprops(ImgBW, 'Area', 'BoundingBox');
for K = 1  : length(stats)
bb = stats(K).BoundingBox;
rectangle('Position',[bb(1),bb(2),bb(3),bb(4)],'EdgeColor','r','LineWidth',2)
barea = bb(3) * bb(4);
a = stats(K).Area;
fprintf('Area #%d width = %d height = %d true area %d\n', K, bb(3), bb(4), stats(K).Area);
end
hold off
```

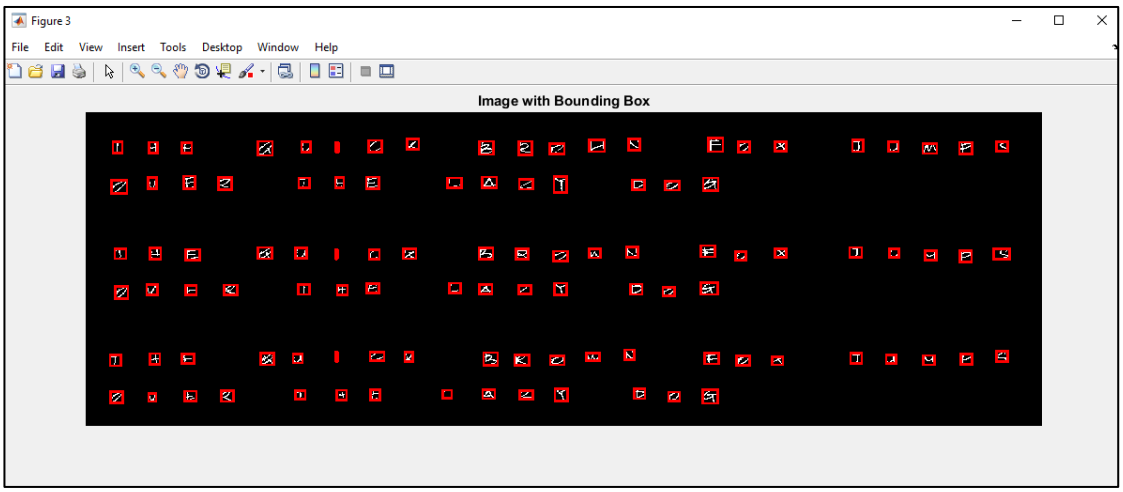
Figure 3.5: Complete MATLAB coding



(a) – the original image (RGB image)



(b) – the black and white image



(c) – the image with bounding box

Figure 3.6: The transformation of the raw image

Thus the bounding box will be formed around the white region of the alphabet as shown in Figure 3.7. The true area indicates the handwritten alphabet area on the survey form. As the alphabet is separated respectively, the true area able to find for each alphabet. The true area is the white region inside the red bounding box. The number of pixels that covers the white region is the true area of the respective alphabet. On the other hand, the bounding box is also being used to identify the height and width of the alphabet as shown in Figure 3.7. The height of the letter is measured from the lowest point to the highest point of the white region on the alphabet. As for the width, the measurement is made from the left most to the right most of the white region on the alphabet.

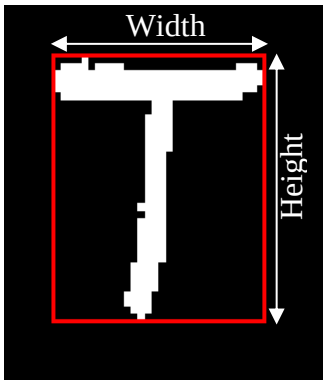


Figure 3.7: Bounding box of an alphabet

There are 35 alphabets in each " THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG" phrase and the phrase is written trice in a data sheet. Therefore, there are 105 alphabets in each data sheet. As there are many alphabets in one data sheet, the numeric labeling is required for the process of identification of height, width and true area of the respective alphabet from the result shown in MATLAB. Each alphabet was labeled from 1 to 105 as shown in Figure 3.8.

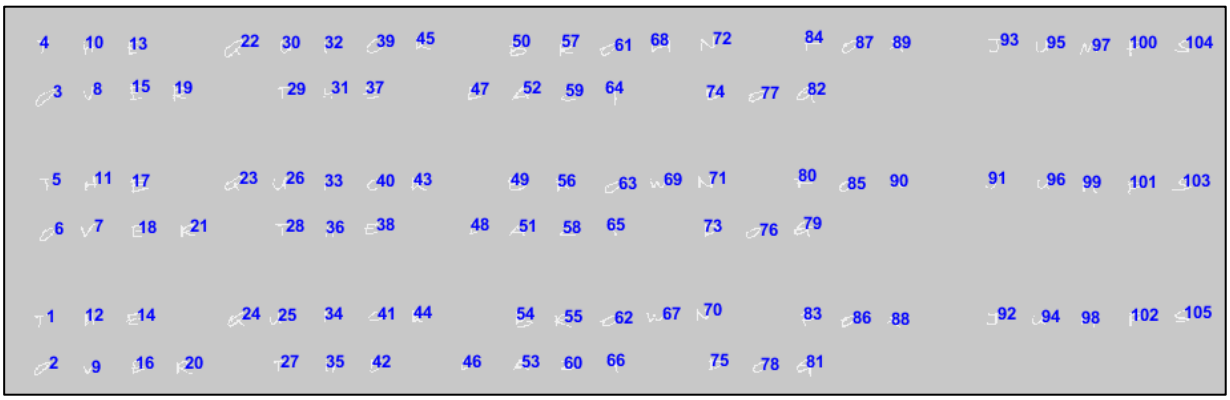


Figure 3.8: Numeric labeling for respective alphabet

The outcomes of the MATLAB coding will be displayed in 105 columns format as shown in Figure 3.9. "Area #1" indicates the numeric label of the alphabet, width, and height represents the letter width and height respectively while the true area represents the white region inside the bounding box. In other words, the true area is the region covered by the written alphabet. The resulting values were measured by pixel units.

```
Area #1 width =    height =    true area
Area #2 width =    height =    true area
Area #3 width =    height =    true area
Area #4 width =    height =    true area
Area #5 width =    height =    true area
Area #6 width =    height =    true area
Area #7 width =    height =    true area
Area #8 width =    height =    true area
Area #9 width =    height =    true area
```

Figure 3.9: Sample format of the Matlab coding outcomes