Short Answer Assessment System with Student Identification using an Automatic Off-line Handwriting Recognition System and Novel Combined Features

Author

Suwanwiwat, Hemmaphan

Published

2016

Thesis Type

Thesis (PhD Doctorate)

School

Institute for Integrated and Intelligent Systems

DOI

https://doi.org/10.25904/1912/2907

Short Answer Assessment System with Student Identification using an Automatic Off-line Handwriting Recognition System and Novel Combined Features

Hemmaphan Suwanwiwat

Master of Information Technology

Institute for Integrated and Intelligence Systems

Griffith Sciences

Griffith University

Submitted in fulfilment of the requirements of the degree of

Doctor of Philosophy

July, 2015

Abstract

Examinations are a widely used form of assessment and thus important in the learning process. However, handwritten examination assessment for certain types of examination, such as essays and short answer questions, is a difficult task; it requires the markers' concentration, precision and it is time-consuming. Off-line automatic assessment systems can be an aid for teachers in the marking process. Handwriting Recognition is one of the most intensive areas of study in the field of pattern recognition. The automatic assessment of exam scripts can benefit from off-line handwriting analysis methods. There has been no recent work in the development of off-line automatic assessment systems using handwriting recognition, even though such systems will clearly benefit the education sector. The reason for this is that many schools and universities in many parts of the world still use paper-based examination.

A complete off-line short answer assessment with student identification system can be an aid for teachers in the marking process as they reduce the time taken by the human marker. For the proposed system, once the automatic marking process is completed, a report on each student's mark is produced according to the name components which the system has identified. The system could reduce the time taken in marking examination papers as well as reducing the problem of mis-transcribing from examination paper to the report which may be caused by a human assessor.

Feature extraction is one of the most important processes of a handwriting recognition system. A good feature extraction technique maps sufficient information from raw data to a classifier with the smallest dimension. Selecting suitable features can compress the useless information of the pattern and maintain the meaningful information. Many feature extraction techniques are available; amongst these techniques, three techniques, namely the Modified Direction Feature (MDF), the Gaussian Grid Feature (GGF), and Water Reservoir Feature (WRF), are known for their high accuracies in off-line handwriting recognition problems. For this reason, these techniques were employed in the proposed systems. In-depth investigation of the techniques were conducted and, as a result, two novel hybrid feature extraction techniques called "Water Reservoir, Loop, Modified Direction Feature (WRLMDF)" and "Water Reservoir, Loop, Gaussian Grid Feature (WRLGGF)" were proposed. Furthermore, the original MDF, original GGF, and the two newly proposed WRLGGF and WRLMDF were enhanced with the aid of centre of mass features. All experiments were performed on newly collected bilingual name components and short answer word datasets.

Investigation of the contour deduction was also performed. The experiments were performed by employing upper and lower contour and loops (Three Images - TI), upper and lower contours (ULC), and Full Boundary Contour (FBC) in the feature extraction process. The feature extraction techniques were performed on the three images without extracting from the full boundary images.

The proposed WRLMDF outperformed the original MDF in a number of experiments by 1.37% – 4.19%. It was also found that in some experiments TI and ULC yielded better results than FBC. The proposed WRLGGF achieved the highest recognition rates of 99.52%, 97.49%, 95.94%, 99.25% and 95.99% of 1,040 sample English SIS, 3,940 sample English SIS, 3,940 sample Thai SIS, 2,060 sample bilingual SIS, and 7,880 sample bilingual SIS, respectively. For the SAAS system, an encouraging recognition accuracy of 95.99% was achieved employing WRLGGF compared to 92.65% for GGF, 88.15% for MDF, and 89.89% using WRLMDF. Furthermore, in short answer word automatic assessment experiments employing a short answer word dataset, improved recognition rates from 0.16 to 6.06% were achieved when employing enhanced MDF, GGF, and WRLGGF compared to the original MDF, GGF and WRLGGF.

The proposed SAAS system was able to recognise and mark examination papers, together with the SIS being able to identify students from their name components successfully. The results produced by the proposed system were encouraging. In the near future, student identification systems can be extended to student identification and verification systems where biometrics are involved which will expand further the usability of the student identification system. The short answer automated assessment system can also be extended into long answer or essay marking systems.

Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due acknowledgement is made in the thesis itself.

____

Hemmaphan SUWANWIWAT

Acknowledgements

Words are insufficient to express the depths of my gratefulness for those who have supported me during the period of my candidature as my feelings are much stronger than any words can explain. First of all, I would like to express my sincere gratitude and thanks to my supervisor, Professor Dr Myer Blumenstein, for his expertise, excellent guidance, support, encouragement, and for being so patient with me. I appreciate all of his contributions which have made my PhD successful. Professor Blumenstein, you are the best.

I would like to give special thanks to my external supervisor, Professor Dr Umapada Pal of the Indian Statistical Institute, Kolkata, India, for his valuable guidance and advice during the period of my PhD study. Having had him as my external supervisor was priceless.

I would like to give thanks to Associate Professor John Thornton, Griffith University, Gold Coast, Australia, for his valuable comments on my PhD confirmation seminar. I also would like to thank Professor Dr Yongsheng Gao for being my supervisor and his important comments throughout the period of my PhD.

I would like to thank Dr Vu Nguyen, Dr Nabin Sharma, Seyyed Adel Alavi Fazel and Ranju Mandal for their ideas, inspiration and support. Having you as my friends and colleagues has really been a privilege, and working with you has always been productive and rewarding. Also I would like to give my thanks to all of my colleagues and friends for their continued support and encouragement.

I also would like to thank Ms Victoria Wheeler, Ms Natalie Dunstan, Ms Lorraine Lauriston for their assistance and friendship. And I would like to give special thanks to Ms Claire Rodway for English proofreading, for being so patient, and for her advice. Thank you also to all the academic and technical staff from the School of Information Technology and to the administrators of the Griffith High Performance Computing.

Finally, I would like to give my wholehearted thanks to my parents and my family for their love, understanding, patience, encouragement and inspiration. This research would not have been possible without their support.

Contents

Contents

Contents

Contents

List of Figures

List of Figures

List of Figures

List of Figures

List of Figures

List of Tables

List of Tables

Nomenclature

AAR	Assessment Accuracy Rate
ACT	American College Test
AEG	Automated Essay Grading System
ANN	Artificial Neural Network
BSIS	Bilingual Student Identification System
CAA	Computer Assisted Assessment
CBA	Computer-based assessment
CH	Cursive Handwritten
COG	Centre of Gravity
COMF	Centre of Mass Feature
CRR	Correctly Recognised Rate
DF	Direction Feature
DP	Dynamic Programming
EE	Essay Examination
EGGF	Enhanced GGF
EMDF	Enhanced MDF
ER	Efficiency Rate
ESIS	English Student Identification System
EWRLGGF	Enhanced WRLGGF
EWRLMDF	Enhanced WRLMDF
FBC	Full Boundary Contour
FET	Feature Extraction Technique
GMAT	Graduate Management Admission Test
HA	Holistic Approach
HMM	Hidden Markov Model
HP	Hand-Printed
HPCHC	Hand-Printed and Cursive Handwritten words Combined
HVBC	Holes, Vertical Bars, and Cups Feature
KLT	Karhunen-Loeve transform
KNN	K-Nearest Neighbour
LF	Loop Feature
LGGF	Loops and Gaussian Grid Feature

Nomenclature

LMDF	Loops and Modified Direction Feature
LSA	Latent Sematic Analysis
LT	Location Transition
MCE	Multiple Choice Examination
MCQs	Marking multiple Choice Questions
MDF	Modified Direction Feature
MLP	Multi–Layer Perceptron
MWRLGGF	Modified Water Reservoir, Loop and Gaussian Grid Feature
NLP	Natural Language Processing
NRR	Non Recognised Rate
OAA	One-Against-All
OAO	One-Against-One
OCR	Optical Character Recognition
OFLAAS	Off-Line Automatic Assessment System
OFSASIS	Off-line Short answer Assessment with Student Identification System
OMR	Optical Mark Recognition
PDA	Personal Digital Assistant
RBF	Radial Basis Function
RR	Recognition Rate
SA	Short Answer
SAAS	Short answer Assessment System
SAE	Short answer examination
SAT	Scholastic Aptitude Test
SIS	Student Identification System
SLP	Single-Layer Perceptron
SURF	Speeded Up Robust Feature
SVM	Support Vector Machine
TF	Transition Feature
TI	Three Images (upper contour, lower contour, and loop images)
TSIS	Thai Student Identification System
ULC	Upper and Lower Contours
WR	Water Reservoir
WRGGF	Water Reservoir and Gaussian Grid Feature
WRLGGF	GGF combined with the Water Reservoir and Loop features
WRLMDF	MDF combined with the Water Reservoir and Loop Feature
WRMDF	Water Reservoir and Modified Direction Feature

List of Publications

Forthcoming Journal Paper:

1. Hemmaphan Suwanwiwat, Umapada Pal, and Michael Blumenstein. An Automated Assessment System for Student Identification via Off-line Handwriting Recognition. *Pattern Analysis and Applications (PAA) Journal,* To be submitted in March 2016

Published Conference Papers:

1. Hemmaphan Suwanwiwat, Umapada Pal, and Michael Blumenstein. Short answer question examination using an automatic off-line handwriting recognition system and a novel combined feature, *In Proceeding of the* 2015 *International Joint Conference on Neural Networks (IJCNN '15)*, Pages: 1 – 8, 2015
2. Hemmaphan Suwanwiwat, Umapada Pal, and Michael Blumenstein. A complete automatic short answer assessment system with student identification, *In Proceeding of the 13th International Conference on Document Analysis and Recognition (ICDAR 2015)*, Pages: 611 – 615, 2015
3. Hemmaphan Suwanwiwat, Vu Nguyen, Umapada Pal, and Michael Blumenstein. Off-line handwritten Thai name recognition for student identification in an automated assessment system. *In Proceeding of the* 2014 *International Joint Conference on Neural Networks (IJCNN '14)*, Pages: 2347 – 2353, 2014.
4. Hemmaphan Suwanwiwat, Umapada Pal, and Michael Blumenstein. Off-Line Handwritten Bilingual Name Recognition for Student Identification in an Automated Assessment System. *In Proceeding of the* 14*th International Conference on Frontiers in Handwriting Recognition (ICFHR '14)*, Pages: 271 – 276, 2014.
5. Hemmaphan Suwanwiwat, Vu Nguyen, and Michael Blumenstein. Off-line restricted-set handwritten word recognition for student identification in a short answer question automated assessment system. *In Proceeding of the* 12*th International Conference on Hybrid Intelligent Systems (HIS '12)*, Pages: 167 – 172, 2012

Ethical Clearance

Granted Ethical Clearance GU Protocol Number: ICT/01/11/HREC

Chapter 1

INTRODUCTION

Handwriting is very important to human communication and culture as it is the most important sign system ever invented apart from speech [1]. Without handwriting the world may not have developed culture and civilisation as it has, as handwriting is a vital way to enhance human memory and also to assist communication even in the digital world of today. Handwriting remains in use because of the convenience of paper and writing instruments [2], and has three elementary characteristics, which are: 1) it consists of artificial graphical marks on a surface; 2) its purpose is to communicate something; and 3) the communication purpose is achieved by virtue of the mark's conventionality relative to language [1].

Handwriting recognition is one of the most intensive areas of study in the field of pattern recognition and consists of on-line and off-line handwriting recognition techniques. Handwriting is personal and individual [2]; therefore, it varies from one individual person to another. This handwriting characteristic is challenging for researchers to overcome, as it is difficult for handwriting recognition systems, both on-line and off-line, to be able to recognise a given word in different individual handwritings. Unconstrained cursive handwriting is the most challenging problem in the field of Optical Character Recognition (OCR) [3].

OCR is used to derive the meaning of the characters and words from their digitised images. OCR methods are usually applicable to recognition of either handwritten or machine printed character [4]. The prologue to OCR was the attempt to recognise printed material with the objective of developing reading machines for the visually impaired. Later, in 1929, the first patent for the purpose of recognising printed characters was attained by Tausheck [5] in Germany. From then until now, handwriting recognition has been intensively and continuously researched.

Examinations are a widely used form of assessment and thus important in the learning process. However, handwritten examination assessment for certain types of examination, such as essay and short answer questions, is a difficult task; it requires the markers' concentration, precision and it is time-consuming. Despite the fact that computer-based assessment systems are now in use worldwide, paper-based assessment systems are still used in many parts of the world. This is due to the fact that in some countries, equipment is not always available and the cost could be considered expensive. Automatic assessment is one of the off-line handwriting recognition applications. Having an automatic handwritten examination assessment system would obviously be advantageous for markers to overcome some of the aforementioned problems.

This chapter is organised as follows. The background is presented in Section 1.1. Computerised handwriting recognisers associated terms are presented in Section 1.2. On-line and off-line input types are defined in Section 1.3. Automatic assessments of exam scripts are described in section 1.4. Section 1.5, 1.6 and 1.7 describe research aims, objectives, research questions, motivation, and original contribution respectively. Lastly, the organisation of the PhD thesis is presented in Section 1.8.

1.1 Background

In 1929, Tauschek [5] proposed the idea of the first OCR, which was used for template/mask matching. Then in 1933, Paul W. Handel obtained a US patent for OCR in the USA [6]. However, it was not until the 1950's when computer technology arrived, that it enabled researchers to develop handwriting recognition systems.

The 1950's up until the early half of the 1960's are said to be the age of trial and error in OCR when despite great difficulties, researchers realised an ideal OCR [7]. Many recognition systems have been developed since then and many of those are commercially available. In 1951 the modern OCR, a robot reader writer called GISMO, was invented by David Shepard. In 1954, Jacob Rabinow then invented a prototype machine which was able to read upper case; however, the machine was considered slow with the speed of a single character per minute [8]. Researchers in the field of handwriting recognition are now attempting to develop handwriting recognition systems which are able to recognise, interpret and identify both machine printed and handwritten scripts with high recognition and accuracy rates with as few constraints as possible.

In the beginning of the 1960's there were a number of commercial OCR systems. The very first generation of these OCR Systems can be characterised by the constrained letter shapes it could read. Even though these were simple methods, they were very effective. The first commercialised OCR was the IBM 1418. IBM was a very active company in developing OCR. As mentioned before, IBM's OCR was also very constrained, and could only recognise a special IBM font [7].

The middle of the 1960's and the early 1970's was the time of the second OCR generation, which can be characterised by hand-printed character recognition capabilities. With the machines in the early stages, only numerals could be recognised. One important OCR system was the IBM 1287, the first famous second generation OCR system, which was exhibited at the World Fair in New York 1965 [7].

Third generation OCR aimed for poor printed quality characters and hand printed characters for a large category character set. These targets were partially achieved between 1975 and 1985. It has been a long time since the first generation in handwriting recognition system. From then computers have been able to recognise, interpret and identify machine-printed and handwritten script [8]; thus, much improvement in recognising handwriting has been made.

However this has not prevented researchers from continuing their intensive search for solutions in handwriting script recognition because of its challenging nature.

1.2 Automated Handwriting Recognition Systems Terms

There are a few of terms that are important for research in the field of handwriting recognition as these terms are used when dealing with recognising handwriting through automated means. These terms are *recognition*, *interpretation* and *identification* [2]. The first term to discuss is recognition as the research described in this thesis investigates an automatic off-line assessment system which is one of the handwriting recognition applications. The term refers to the act of transforming the graphical marks associated with human handwritten script into symbolic representation that are stored on a computer system in the form of 8 bit ASCII code or 16-bit Unicode which are both currently in use.

The second term is interpretation which refers to the task of determining the meaning of a section of handwritten text. One good example of this is the interpretation of a handwritten address on an envelope.

The last term is handwriting identification which refers to the task of authorship identification. This task is based on an assumption that each person's handwriting is individual. A good example of the task of handwriting identification relates to signature verification. Another good example of an identification and verification application is in forensics, when identification of an individual's handwriting is required. Handwriting recognition and interpretation in forensics are, additionally, processes with the objectives of filtering out variations so as to determine the message.

Identification is very different from recognition and interpretation. This is because while identification attempts to identify the unique nature of the portion of handwriting, to differentiate two writers for instance, recognition and interpretation require the elimination of the handwriting variations from the portions of handwriting. By doing so it allows the handwriting to become uniform and the meaning of the message to be extracted more easily [8].

1.3 Automatic Assessment of Exam Scripts

The automatic assessment of exam scripts [9], [10], [11], [12], [13], [14] is one of the applications that benefits from off-line handwriting analysis methods. Examination assessments are divided into two categories, which are summative and formative. Whereas summative assessment takes place at the end of a teaching period in order to evaluate the students' sum of knowledge, the formative assessment is an iterative process used to evaluate the students understanding of a subject while teaching is still in progress. The formative assessment is generally simpler to facilitate, and therefore can be assessed in a shorter period of time [15]. A summative form of assessment will be used in developing the proposed system.

Figure 1.1 below shows a block diagram of a general off-line examination assessment system. The system consists of examination scripts acquisition, preprocessing, feature extraction, classification, and assessment phases. Lexicon usage is typically employed in handwritten word recognition by using it for the possible words for the answer to the exam answers [13], [9]. The images of examination papers and name component examples can be seen in Figure 1.2 (a) and (b).

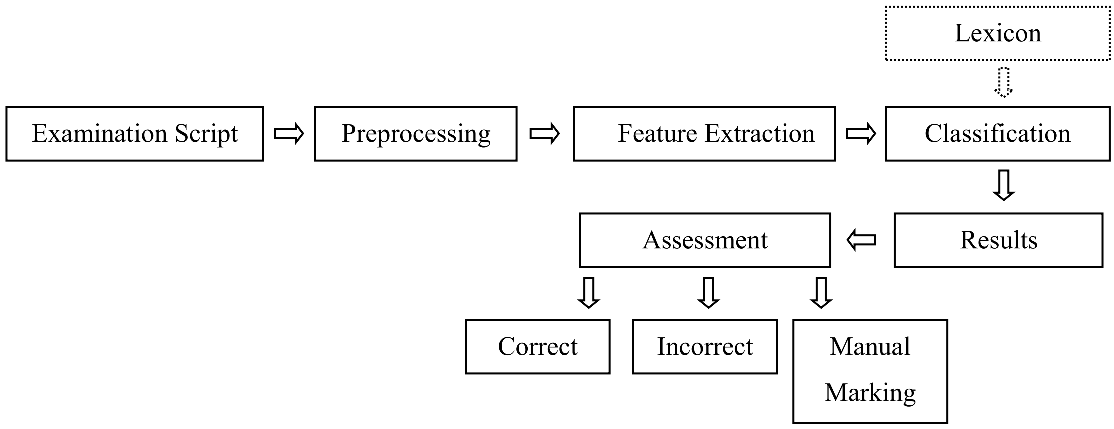


Figure 1.1: A basic block diagram of a general off-line examination assessment system

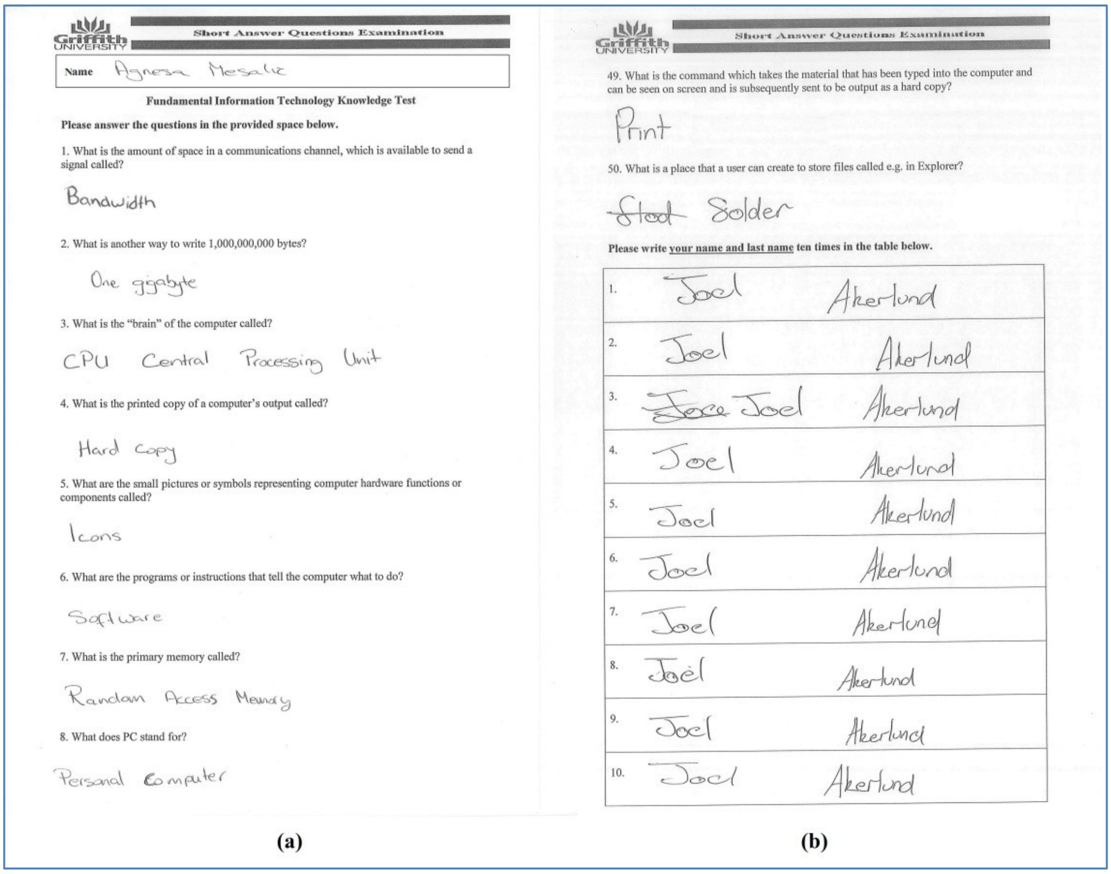


Figure 1.2: Examples of examination papers employed in this research (a) exam questions and answers (b) name component examples

1.4 Aims and Objectives

The automatic assessment of exam scripts can benefit from off-line handwriting analysis methods. To the best of the author's knowledge, there are not many studies which have been done with off-line short answer automated assessment systems, and especially not with a student identification system which employs student name components in its identification process.

The objective of this thesis is to create hybrid feature extraction techniques which can be used in an Off-line Short answer Assessment with Student Identification System (OFSASIS), to extract the salient information that needs to be applied in the recognition process. The successful system could reduce the time taken in marking examination papers as well as reducing the problem of mis-transcribing from examination paper to the report which may be caused by a human assessor. For the OFSASIS, once the automatic marking process is completed, a report on each student's mark is produced according to the name components which the system has identified.

In this research, short answer examinations are those in which examinees can freely write anything they believe to be the correct answers to the questions set. As there are minimum constraints on the style of writing, writing instruments, and answers to be written, it is more difficult to recognise all the possible words that examinees may write. To be able to recognise or to mark the answers correctly therefore is a great challenge as the cost of marking the answers incorrectly is too expensive. Also this is an area with many unresolved problems and such systems will clearly have benefits for education. To overcome these unresolved problems is very challenging.

The main aims and objectives of this research are:

- To design and investigate novel hybrid feature extraction techniques to be used in an automatic system for assessing off-line handwritten short answer exam questions and student name identification. And to improve the recognition rate compared to existing systems. In order to do so, this research proposed and employed:
    a) The Modified Direction Feature (MDF) combined with the water reservoir and loop features (WRLMDF).
    b) The Gaussian Grid Feature (GGF) combined with the water reservoir and loop features (WRLGGF).
    c) Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs) were used as classifiers in the proposed systems.
    d) Centre of mass feature was employed in order to enhance the hybrid feature extraction techniques in a) and b) as well as the original MDF and GGF.
- To improve the automatic assessment system usability. An answer to a question is not always completed and involves more than a simple yes or no. In order to develop a system that is more likely to be used in the real world, marking criteria were added to the

system. By having marking criteria, the usability of an automatic system for assessing off-line handwritten short answer exam questions will be increased.

- To develop the Student Identification System (SIS) and investigate the accuracy of the proposed hybrid feature extraction techniques when applied to the SIS.
- To investigate the proposed hybrid feature extraction techniques ability and accuracy when applying to the bilingual script namely Thai and English languages.

1.5 Research Questions

The research questions which are answered through this study are:

- Is the water reservoir, loop, and centre of mass feature combined with a MDF extraction technique going to yield a better recognition rate than the original MDF extraction technique on its own?
- Is the water reservoir, loop, and centre of mass feature combined with GGF extraction technique going to yield a better recognition rate than the original GGF extraction technique on its own?
- Is the water reservoir feature extraction technique, which was first applied to word images in order to find segmentation points, going to be able to be usable to extract features to be used for recognition process?
- Is the MDF technique, which was first developed to be used to extract features from characters, going to be able to be used to extract features from whole word images efficiently?
- Is the GGF technique, which was first developed to be used in signature verification problems, going to be able to be used to extract features from whole word images?
- Are the proposed hybrid feature extraction techniques and the original MDF and GGF going to be able to be used efficiently in student identification problems using handwritten name components in the bilingual scripts?

1.6 Motivation

Examinations are a widely used form of assessment and thus are important in the learning process. Examinations help teachers to measure their students' knowledge, skills, and development and thereby plan for their students' learning in the future. However, handwritten examination assessment is a difficult task; it requires the markers' concentration, precision and it is time-consuming.

Many schools and universities in many parts of the world still use paper-based examination. Paper-based examination is still in use worldwide despite the fact that computer based examination is now popularly in use. This is due to the fact that not every country in the world is able to afford a sufficient amount of computers for students per class because of potentially high costs, especially in developing countries. To the best of the author's knowledge, there are only a few studies undertaken in developing automatic assessment systems using handwriting recognition, even though a successful system would undoubtedly benefits the education system as schools and universities in many countries still employ paper-based examinations. Additional to the best of the author's knowledge, there is no existing work on an automatic off-line short answer assessment system comprising a student identification component.

Off-line short answer assessment with student identification systems can be an aid for teachers in the marking process as they reduce marking time. Recently developed off-line short answer assessment systems have not been able to operate under unconstrained conditions and still achieve encouraging results. Recently developed systems in this area achieved assessment yield of 54 % with 99% accuracy with heavy constraints [9].

1.7 Original Contribution

This thesis describes heuristic word recognition employed in the proposed Off-line Short answer Assessment with Student Identification System (OFSASIS) using hybrid feature extraction techniques. The major, original contributions are:

- New novel hybrid feature extraction techniques which were employed on newly collected bilingual name components (first, middle, last, and second last name), namely:
    a) A novel hybrid water reservoir, loop, and modified direction feature (MDF) extraction technique (WRLMDF) to improve recognition and accuracy rates in whole word recognition problem.
    b) A novel hybrid water reservoir, loop, and Gaussian grid feature (GGF) extraction technique (WRLGGF) to improve recognition and accuracy rates in whole word recognition problem.
- Enhanced original feature extraction techniques namely MDF and GGF as well as proposed feature extraction techniques being WRLMDF and WRLGGF.
- An off-line short answer assessment with student identification system (OFSASIS) using hybrid feature extraction techniques on heuristic word recognition. These hybrid feature extraction techniques were employed on a newly collected short answer word dataset.
- Increased the efficiency of the proposed and original feature extraction techniques applied to different classifiers, being artificial neural networks and support vector machines.

1.8 The Organisation of the Remainder of this Thesis

The remainder of this thesis is organised into the following four chapters. Chapter 2 consists of a literature review which provides an in-depth of up-to-date handwriting recognition research and off-line automatic assessments/examination systems. Chapter 3 discusses the proposed hybrid feature extraction techniques as well as the methodology for developing a hybrid off-line automatic assessment with a student identification system. Chapter 4 provides the results obtained following the investigation of the proposed hybrid and enhanced feature extraction techniques and the OFSASIS. Chapter 5 offers deep analysis and results comparison to the results from other researchers. Chapter 6 draws conclusions concerning the investigation. Recommended directions for further work are also presented.

Chapter 2

LITERATURE REVIEW

The main focus of this research is developing a hybrid automatic short answer assessment with student identification system that recognises off-line handwritten words. However, it is essential to understand other related research that concerns handwriting recognition. This chapter describes the image preprocessing, and feature extraction techniques found in the literature as they are very important to handwriting recognition systems. Exploration in the areas of numeral, character and word recognition, and details of segmentation and heuristic based word recognition are also described. Since this research also proposes a student identification system, a brief review of handwriting identification and verification is also presented. Some main applications that benefit from off-line handwriting recognition are also included in this chapter. Finally, the employment of intelligent techniques which have been employed in handwriting recognition are described.

This chapter is organised as follows. Firstly, Section 2.1 gives an overview of handwriting recognition and Section 2.2 describes a general off-line handwriting recognition overview. Details of off-line handwriting preprocessing, including binarisation, image resizing, slant estimation and correction, noise removal, slope normalisation, and skeletonisation/thinning are described in Chapter 2.3. Section 2.4 describes feature extraction techniques found in the extant literature. Section 2.5 deals with handwriting recognition, including types of handwriting styles, numeral and word recognition. Off-line handwriting applications are described in Section 2.6. Details on automatic assessment systems, examination types, and existing both on-line and off-line automatic assessment systems can also be found in this section. Section 2.7 gives details on classification in which classifiers including ANNs, SVMs, Hidden Markov Models (HMMs), and hybrid are described.

2.1 Handwriting Recognition Overview

Handwriting recognition can be divided into two different forms, which are off-line and on-line. These two forms are very different; hence it is necessary to know each form's quality. The first type is the on-line handwriting recognition system. This system recognises handwriting in real time, which means it recognises the writing as the writer is writing [16]. Devices such as a digitiser or an instrumented stylus are used to capture the writing information during the act of writing [2]. On-line recognition relies on information acquired during the production of the handwriting such as coordinates of the writing trajectory, pen pressure and time. Mobile communication systems such as Personal Digital Assistant (PDA), electronic pad and smart phone have intergrated on-line handwriting recognition capabilities [17]. Handwriting data is captured by means of an electronic tablet or writing device. This data capture has several advantages as it provides the temporal information of pen-tip trajectory which is an additional source of knowledge that helps to increase the recognition accuracy. The information also includes the number of strokes, their order, the ordered sequence of coordinate points, and writing speed. The ordering of strokes, which constitute a word, and is useful in segmenting a word into characters, is encoded [18], [19]. This allows the sequence of sampling points along the writing trace to be known. On-line systems are usually user adaptable, which facilitates the recognition process [18]. There are many applications that benefit from on-line handwriting recognition, for example whiteboard notes recogniser [19], on-line signature verification, mobile devices, PDA [20], and automatic transcription of multilingual documents [21].

The second type is off-line handwriting recognition, which in contrast, recognises the written document. By using a scanner, the hard copy document is transformed into a digitised pattern [16]. This allows the off-line handwritten recognition system to subsequently recognise a binarised handwritten image. A binary raster image is the only input and the systems must perform a user independent recognition process [18]. Off-line or static recognition systems rely on more sophisticated architectures than the on-line recognition systems when trying to accomplish the same recognition task. However, when compared with the recognition results with those from on-line systems, the recognition rates are still less. This is due to the lack of an additional source of knowledge which is available in the on-line systems [22]. Off-line handwriting techniques also have other disadvantages compared to on-line handwriting techniques. One main disadvantage is that the on-line technique captures real time information of the writing which is important in the recognition process, including order of the strokes, each stroke direction, and the speed of the writing within each stroke, whereas the off-line technique does not.

It is to be noted that some on-line information can be extracted from off-line text by exploiting the constraints governing human hand movement. A method proposed by Govindaraju *et al.* [23] was used to derive the temporal information from static cursive word images in the form of the upstrokes and downstrokes of each stroke. There are also many applications that benefit from off-line handwriting recognition, including handwritten address interpretation, bank cheque recognition, writer identification, and signature verification are examples of off-line handwriting techniques.

2.2 General Off-line Handwriting Recognition System Overview

Generally, off-line handwriting recognition systems consist of 1) data acquisition from raw document usually by using a scanner, 2) preprocessing including background subtraction, thresholding, noise removal, filling, stroke restoration, slant normalisation, skew normalisation, baseline estimation, segmentation, and skeleton extraction, 3) feature extraction, 4) recognition /

classification process. A basic block diagram in Figure 2.1 illustrates a general off-line handwriting recognition system.

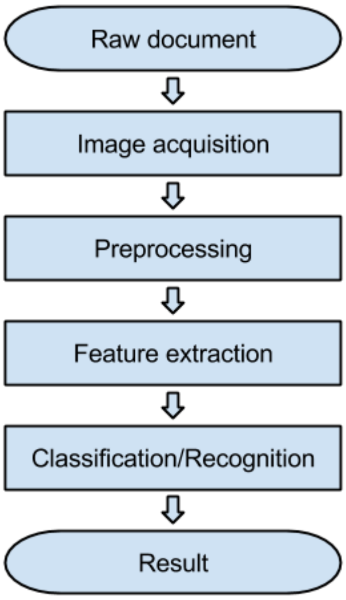


Figure 2.1: A basic block diagram of a general off-line handwriting recognition system

2.3 Preprocessing

Image preprocessing is a necessary operation which has to be performed before the recognition process. The image preprocessing processes are performed in the very beginning of document analysis. The main purpose of the image preprocessing is to develop useful canonical descriptions of shapes and surfaces in the given image [24]. In the other words, preprocessing enhances the actual image for suitable further analysis [25]. Output from these pre-processes will be used as input for the recognition phase; hence, the best image preprocessing techniques will be integrated in order to achieve the best valid output which will result in higher recognition and accuracy rate.

The preprocessing includes many tasks: the usage of each task depends on the requirements of subsequent processes. The tasks include, but are not limited to image acquisition, thresholding, noise removal, line removal, filling, smoothing, stroke restoration, slant normalisation, skew normalisation, upper and lower baselines detection, segmentation, boundary extraction and skeleton extraction. This section describes some of the aforementioned preprocessing techniques.

2.3.1 Image Acquisition

Image acquisition is the very first state of handwriting recognition. Handwritten paper, which has been written on sometime in the past, needs to be converted into digital form (i.e. in the form of a binary or grey-scale image) [26]. This process is usually done by using a scanner.

Whether the scanned image quality is good or poor depends on the resolution used, quality of the scanner hardware and calibration of the scanner [27]. Vargas *et al.* [28] reported the influence of image resolutions on their off-line signature verification system, using the Otsu algorithm [29]. The resolutions experimented were from 45 dpi to 600 dpi; they found that for their particular signature verification system, despite higher computation cost, there was no substantial improvement attained when applied the resolution greater than 150 dpi was applied. In the literature, the resolution which is commonly used is 300 dpi [30], [31], [32], [33], [34], [35]. Figure 2.2 illustrates a basic block diagram of image acquisition process.

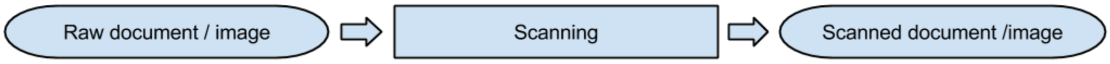


Figure 2.2: A basic block diagram of the image acquisition process

2.3.2 Thresholding/Binarisation

The scanned images are primarily stored in a grey-level format. Each image's pixel intensity may vary between a value of 0 and 255. The value of 0 indicates a white pixel, and the value 255 indicates a black pixel. As the value of each pixel can vary in shades of grey, the conversion of the primarily grey level image into a less storage intensive format of black and white has been applied [8]. There is controversy over whether or not the recognition performed on features directly extracted from grey-scale produces better results than those from binary images. Whether the researchers decide to employ binarised images or not depends mainly on the feature extraction techniques employed afterward. However, many document image analysis researchers have used the binarised images in their systems [27], [36], [37].

Binarisation or thresholding is a process of converting a grey-level image to a binary image. Image binarisation is of great importance in the document analysis and recognition pipeline because it affects further stages of the recognition process [37]. This operation selects elements of an image based on whether they may be considered background (white pixels) or foreground (black pixels or parts of the word image itself). A threshold is usually applied so that pixels with a luminance over the threshold are marked as being background pixels, and the foreground pixels are those with a luminance under the threshold [8]. Hence selecting an appropriate threshold is important.

Trier *et al.* [27] explain that binarisation method may be divided into two categories which are global and locally adaptive binarisation methods. For global binarisation, a single threshold value is calculated from a whole image. The pixels which have a grey level darker than the threshold value are considered foreground (black pixels); otherwise they are considered as background (white pixels). It may be noted that Otsu's [29], and Kittler and Illingworth's [38] global binarisation methods are known to be optimal under normality assumptions. On the other hand, locally adaptive binarisation methods compute a threshold for each pixel on the basis of information contained in a neighbourhood of the pixel. Trier *et al.* [27] evaluated thresholding methods both global and locally adaptive binarisation methods by applying them on map images. The map images were low in contrast and had variable background intensity and noise. They reported that locally adaptive binarisation outperformed the global methods [27].

Even though the locally adaptive performances were better than the global ones, it may be noted that the best locally adaptive method [39] took 10 times longer to execute than the best global method [29]. An interesting conclusion from their evaluation is that based on contaminated and chipped away pixels from the foreground images, certain global methods ranked relatively well. However, some of the best locally adaptive techniques still did not yield adequate binarisation accuracy for use in further automatic processing. This might be an explanation as to why some researchers still prefer to extract features directly from grey-level images [8], [40], [3], [41], [42], [43]. Binarised images use less storage, and manipulating them is less expensive than the grey scale images. However, for binarised images, loss of information, distortion and noise could occur.
Thresholding techniques can further be divided into four subcategories [44], which are histogram-based methods, clustering-based methods, object attribute-based methods, and discrimination based on local pixels' characteristics.

1) Histogram-based methods are regarded as global thresholding methods which can perform well when applied to a clear bimodal image. They can be divided into two sub-types: histogram entropy-based algorithms and histogram shape-based algorithms. The entropy-based algorithms consider certain measures of both entropy of the original image and that of the binarised image. The shape-based algorithms can be used to determine the threshold level for the binarisation of the image. In these methods, peak detection, valley-seeking threshold selection, and histogram concavity analysis, for example, are used. The local histogram analysis methods have been proposed for the images with non-uniform background or those which are degraded or noisy. These methods divide an image into different zones through layout analysis. By doing so, the pixels in a zone are homogeneous. However, the window size used in the approach has to be determined, and the block effect overcome.

2) In clustering-based methods, grey level samples are clustered in two parts as background and foreground. The samples can also be modelled as a mixture of two Gaussians iterative thresholds. The well-known Otsu's method is one of these methods.

3) Object attribute-based methods search for a measure of similarity between the grey level and the binarised image.

4) Discrimination is based on local pixel's characteristics. For these techniques, a threshold is computed at each pixel. The pixel depends on some local statistics such as range, variance, or surface-fitting parameters of the neighbouring pixels.

It should be noted that Otsu's thresholding method is the most referenced thresholding method [44]. Many researchers have been trying to improve the efficiency of Otsu's method of multi-level thresholding problems, which include the inefficiency in determining the best thresholds caused by the fact that it involves a great number of repetitive computations of zero and first order cumulative moments of the grey level histogram [45]. An example of a scanned and its binarised images can be seen in Figure 2.3.

(a) **(b)**

Figure 2.3: (a) Scanned image and (b) its binarised image

Researchers [46] have stated that based on character recognition one main reason that can lower the recognition rate is caused by the sensitivity to the deformation of the image of a character. The well-known factors that cause such a deformation are:

- Noise: Noise is generally introduced by the optical scanner, the writing surface or the writing instrument. Noise can cause, for example, disconnected line segments, bumps and gaps in lines, and filled loops.
- Distortion: Distortion includes, for example, local variations, rounding of corners, improper protrusions, dilation and shrinkage. Distortion is mainly produced by the writer.
- Style variation: Style variation is the use of different shapes to represent the same character, as well as serif and other flourishes, and slant, for example, and is generally produced by the writer.
- Translation: Translation is the movement of the whole character or its components. Translation may result from both the writer and the mechanical part of the recognition system.
- Rotation: Rotation is the change in orientation, which may result from both the writer and the mechanical part of the recognition system.

As mentioned above, deformation and image artefacts can lower the recognition rate. In order to minimise image deformation and artefacts caused by scanning and binarisation, some important operations can be applied. These operations include resizing, slant estimation and correction, noise removal, slope normalisation, and skeletonisation.

2.3.3 Noise Removal and Stroke Repair

As explained by L O'Gorman [4], noise is generally introduced after images have been scanned and binarised. Salt and pepper noise is a common artefact in poor quality image. Salt and pepper noise appears as isolated pixels or pixel regions of ON noise in OFF backgrounds or OFF noise within ON regions and as rough edges on characters and graphics components. Some irregularities such as touching line segments and smeared images can be found [8]. Noise removal process is used to filter, remove, or reduce noise. It is an important process that helps to reduce or remove the ambiguous features and therefore improve recognition rates. In addition, noise reduction reduces the size of the image file, and hence reduces the time required for the subsequent processing and storage [4].

Suen *et al*. [46] suggest that an important pre-processor function is to remove isolated pixels and bumps while also filling holes, and refers to this process as smoothing. Smooth contour approximation is needed to reflect the general orientation of the strokes [18]. In one study [47], an averaging technique was used as a smoothing technique. Smoothing is normally done by considering the bit patterns in small areas called windows (e.g. an area of 3 x 3 matrix elements [46]). Erosion and dilation techniques can be used to reduce or remove noise [48]. Erosion is a shrinking operation of the object while its dual operation dilation can be regarded as the expansion of an object.

(a) (b)

Figure 2.4: (a) Noisy image and (b) noise removed image

The other important pair of operations after erosion and dilation are opening and closing. In brief, opening is erosion followed by dilation, while closing is a dilation followed by erosion. The opening can be used as a filter to smooth the contours and suppress small islands (noises), while the closing can block up narrow channels and thin lakes [48]. An example of noisy and noise removed images can be seen in Figure 2.4.

Binarisation may introduce noise as well as introduce broken strokes; as a result, stroke restoration may be required. Stroke reconstruction involves "filling" in gaps in the broken strokes. An adaptive stroke repairing algorithm was proposed by Shi and Govindaraju [49]; their algorithm involves selective and adaptive stroke "filling" with a neighbourhood operator which emphasises stroke connectivity. Further investigation on stroke restoration, which was done on signature verification problems, was proposed by Siyuan and Srihari [50]. A larger selective region of 7 × 7 was employed; they found that foreground pixels separated by further chessboard distance can be connected.

2.3.4 Scaling / Resizing

Scaling or resizing is the process that makes the image invariant to size. It transforms an input image of arbitrary size and output image of a fixed pre-specified size with an attempt to preserve the original structural details [51]. Resizing may be necessary when using some certain classifiers, for example Artificial Neural Networks that require the feature vector to be uniformed [8]. There are various types of scaling techniques used in the literature [40], [51]. For character size normalisation, the most popular method is linear normalisation. This technique bounds the character strokes with a rectangle and linearly maps the rectangle into a standard size. The use of moment normalisation can increase recognition accuracy. In this technique, the centre of gravity of the character image is in alignment with the geometric centre of a normalised plane, and the x-axis and y-axis scales are determined by second-order moments. The second-order moment resets the boundary of character for aligning with the square of normalised plane [40].

A simple scaling method can be applied on binarised images. This can be done by taking a bound boxed image and scaling it proportionally to a specified size. A direct pixel mapping is used to generate the normalised image [51]. A standard size normalisation technique can be performed by modifying the Cartesian coordinates of the input image by multiplying each via scaling constants. Reverse mapping is performed to decide how the coordinates shall be filled in the destination image. A problem may occur when the mapping function calculates a fractional pixel address. This problem may be solved by applying an interpolation technique, which estimates the new pixel in the destination image by obtaining a value from some function of the neighbours of the corresponding source address. Some examples of interpolation methods are nearest neighbour and bi-linear interpolation [8].

2.3.5 Slant Normalisation

The next preprocessing process is slant estimation and correction. Slant is an individual variation in handwritten words. To reduce effects of slant variation, the slant has to be estimated and corrected. The slant normalisation is based on the hypothesis that, when the slant is minimal, the number of vertical strokes is maximal. A measure of the number of vertical strokes is given by the number of columns where there are no background pixels between the highest and lowest foreground pixel [52].

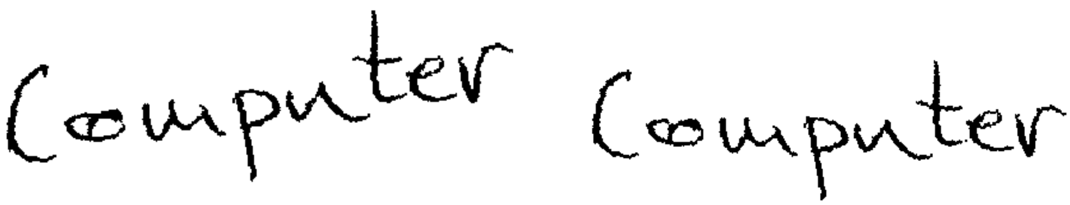
A technique used by Caesar *et al*. [18] was shear angle normalisation which was performed on the contour level of the Binary Connected Components (BCC). It was also stated in [52] that after having applied a shear transformation to the original image for each angle at a reasonable interval, the value of the sum is obtained. The angle corresponding to the shear transformed image that gave the highest sum is then used as slant estimate. A comprehensive slant estimation and correction technique was described in [53]. In some applications, such as writer dependent recognition, slant correction and height normalisation are not absolutely necessary in a writer dependent system [54]. An example of a slant image and its slant normalised image can be seen in Figure 2.5 (a) and (b), respectively.

(a) (b)

Figure 2.5: (a) Slanted image and (b) slant normalised image

2.3.6 Skew / Slope Normalisation

Brakensiek [55] states that a necessity for feature extraction is an estimation of baseline, which is done by an approximation of a horizontal line to the local minima of the word respectively by sentence. Furthermore, baseline and top minuscule line estimation are essential for rotation normalisation, vertical scale normalisation, and ascender and descender detection. It is more difficult to calculate the top minuscule line because more ascenders exist that disturb the adaptation of the model. For baseline and top minuscule estimations, the contour projections with different thresholds are used [18]. In Bunke *et al.*'s work, four reference lines - the lower line, lower baseline, upper based line and upper line were used to retrieve features representing information about the location of an edge relative to these four reference lines, its curvature and the degree of the nodes incident to the considered edge. The horizontal projection was used to identify the reference lines [56].



(a) (b)

Figure 2.6: (a) Skewed image and (b) skew normalised image

Generally when papers are scanned, they are not perfectly aligned with the coordinate of the scanner system [57]. Furthermore, in most unconstrained handwriting, baseline skew is present to varying degrees [58]. In order to resolve this problem, skew or slope normalisation methods are implemented. Some skew angle estimations are based on horizontal projection profile, contour [53] and other quantities [57]. As described in [52] slope/skew normalisation technique consists of finding first a rough estimate of the core region, then in using the stroke minima close to its lower limit to fit the lower baseline. The image is then rotated until the lower base line is horizontal. The first estimate of the core region is fundamental and is found by thresholding the distribution of the horizontal density (number of foreground pixels per line) values. The use of the distribution (instead of the density histogram itself) is preferred because the noise introduced by local features (e.g. long horizontal strokes in ascenders or descenders) of the word becomes statistically irrelevant in it. An example of a skewed image and its skew normalised image can be seen in Figure 2.6 (a) and (b), respectively.

2.3.7 Thinning or Skeletonisation

In general, when allowing unconstrained writing instruments to be used, stroke width variations across different handwriting samples can occur. In order to reduce the amount of these variations, thinning or skeletonisation algorithms are applied. Thinning or skeletonisation algorithms aim to normalise each stroke into one pixel stroke width without changing the pattern or stroke topology [57].

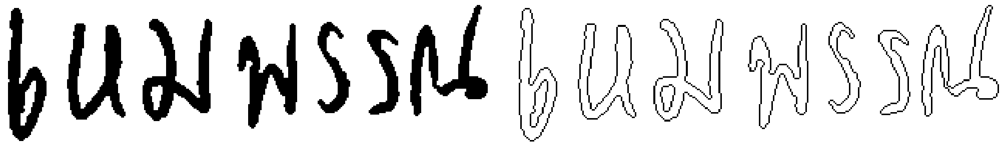
Thinning line analysis has been intensively studied as can be found in the literature. It is one of the very important approaches and many OCR systems have been made based on it. Thinning initiative was proposed in 1957 by Kirsch. Later on, Deutch attempted to use it for character recognition. In 1969, Hilditch proposed the first systematic and rigorous thinning algorithm. There are more than 30 variations of thinning algorithms which have been proposed since then [7]. Problems with the thinning occur as the local operations are used to obtain thinned lines. These problems include imperfect lines for the real lines which include acute corners and/or intersections, and since it is an iterative process, it may take a long time to complete.

A binarised image may be thinned or skeletonised by a standard thinning algorithm. Some artefacts introduced by thinning as stated in [23] are loops and holes in the binary image which are often reduced to single segments and strokes and which intersect on the sequence of points instead of a single point. By using a sample set of rather small stroke width (approximately 3 pixels), a skeleton algorithm may achieve sufficiently good results. The skeleton calculated coincided in most areas with the centreline of the stroke trace [18]. Directional features extraction could be done by employing Kirsch masks [57], and so as an edge detector [59] .

2.3.8 Boundary Extraction

Boundary extraction is one of the preprocessing processes in character and word recognition where its outer and inner contours are found [60]. A number of researchers indicated that handwriting recognition could rely on boundary features based on the fact that humans are capable of recognising different objects from their boundary information. A boundary is specified by the x- and y-coordinates of its pixels which are called boundary pixels.

A comprehensive boundary technique [61] explains that to extract the boundary, first, a rasher scan is used to identify every inner pixel, specifically the foreground pixel that borders four other foreground pixels. Secondly, the identified pixels are assigned the same value as background pixels; this makes the remaining black pixels structure the boundary. Figure 2.7 illustrates an example of an original binarised and its boundary extracted image. Some of the studies which extracted information from boundary images can be found in the literature [50], [62], [63], [64].



(a) (b)

Figure 2.7: (a) Original binarised and (b) its boundary extracted image

2.4 Feature Extraction

Feature extraction is one of the most crucial processes of a handwriting recognition system. The objective of feature extraction is to extract the salient information that needs to be applied in the recognition process. It reduces data dimensionality by determining certain feature properties that distinguish input patterns [16], [24]. A good feature extraction technique maps sufficient information from raw data to a classifier with the smallest dimension [65]. Selecting suitable features can compress the useless information of the pattern and maintain the meaningful information [59]. There are many feature extraction techniques available; the important task is to implement the technique that is suitable for the system.

Feature extraction can be divided into 3 categories which are global, distribution of points, and geometrical and topological features. The first category is global features; a global feature is a statistical feature that represents the shape of the character in an overview [65]. As explained based on character recognition by Suen *et al.* [46], global features do not reflect any local, geometrical or topological properties of the drawing itself, but extract the features from every point which lies within a rectangle (i.e. frame) circumscribing the character. Global features can be extracted easily and are essentially unaffected by minor local changes. Techniques that are used to extract global features include:

- **Template Matching and Correlations:** Template Matching and Correlations uses features such as the state of black or white of all those points which lie within the frame. This technique simply measures the similarity between the input character and the stored references by matching and correlating points or groups of points in the frame. However, one disadvantage with this technique is the high dimensionality of the resulting feature vector [46].
- **Transformations and Series Expansions:** Transformations and Series Expansions technique such as Fourier, Walsh, Haar and Hadamard or Karhunen-Loeve series expansions can reduce the high vector dimensionality and extract features invariant to some global deformation (e.g. global translation or rotation). Series Expansions and

transformations provide some freedom from either translation or rotation [46]. It should be noted that the main problems of all global techniques are their dependence on the position alignment and high sensitivity to distortion or style variations. Even a small spot of dirt or noise may displace the frame and seriously affect the positions of the features and so the recognition rate.

The second category is statistical distribution of points which are used to obtain features. The feature set obtained by these methods are smaller in dimension compared to the global feature ones. Statistical features examples which are obtained from statistical distribution of points are zoning, projection histograms or direction histograms [66]. As there are different types of distribution available, some of them are described here:

- **Zoning:** in zoning the frame containing the character is divided into several overlapping or non-overlapping zones and the densities of points in these different regions form the features.
- **Moments**: the moments of black pixels around a chosen centre, for example, the centre of gravity of the character or a chosen coordinate system, are used as features. Moments are known to be size, orientation, rotation and translation independence [67].
- ***n*-tuples**: in *n*-tuples, the occurrence of black or white elements or joint occurrences of these elements are used as features.
- **Characteristic Loci**: vertical and horizontal vectors are generated for every white point in the background pixel in an image. The number of times (a maximum of 2 was used in order to limit the dimensionality of the feature set) the line segments are intersected by these vectors is used as features.
- **Crossings and Distances Features:** these are measured from the number of times line segments are traversed by vectors in specified directions or the distances of elements or line segments from a given boundary such as the frame which contains the character [46]. The examples of directions or angles are 0°, 45°, and 90°. These distributions of points techniques tolerant to distortions and small stylist variations are achieved because some features do take into account some sort of topological information. They also provide high speed and low complexity for implementation. However, the drawback of these techniques is that it is difficult to make masks for them using zoning, *n*-tuples, crossings and distances. The reason is the large number of possible combinations of features and the dynamic variations of size and shape of characters [46].

Trier *et al*. [33] stated that in order to obtain the best feature extraction performance, the technique should be suitable with output from the preprocessing phase. Feature extraction techniques such as unitary transforms, zoning, geometric moment, and Zernike moments are suitable for grey scale sub-images. For binary images, feature extraction techniques such as projection histograms and zoning and Fourier descriptors are suitable techniques for the image type. With skeletonised images, feature extraction techniques such as discrete feature extraction, zoning and Fourier descriptors are suitable. The researchers [33] further stated that it is necessary that the extracted feature match the classifier requirements. For example, discrete features are suitable for decision trees, while real valued feature vectors are ideal for statistical classifiers. Graph or grammar based descriptions are best with structural classifiers. Hybrid classifiers are excluded as multiple features may be extracted from images to suit classifiers.

The third category is geometrical and topological features. Structural features are based on characters topological and geometrical properties, for instance, strokes and their directions, end-points or intersections of segments and loops. Geometrical and topological features concern the features which describe the interesting geometry or topology of the drawing. The features may represent global and local properties of the character. The main advantage of these features is their high tolerance to distortion and style variations and furthermore they can tolerate a certain degree of translation and rotation. Some geometrical and topological features extracted include strokes and bays in various directions, end points, intersections of line segments, and loops. Also, in connection with contour analysis, stroke relations, angular properties, sharp protrusions and intrusions have also been used [46], [68]. A two stage recognition system was proposed by Methasate *et al.* [65]. The system first uses the global features for rough classification to separate the character to group similar shapes. The second stage then uses both global and local features for fine classification to recognise the character within the group.

This research concerns holistic word recognition which applies Artificial Neural Networks (ANNs) as the recogniser as well as Support Vector Machines (SVMs). Therefore, feature extraction in the holistic approach and feature extraction in both ANNs and SVMs will be discussed here.

2.4.1 Unitary Image Transform Features

A unitary transform can be applied to character images to obtain a reduction in the number of features while most of the information about the character shapes is preserved [33], [66]. As features which are extracted from the images are non-rotation invariance, the input images have to be rotated to a standard orientation. It must be noted that all images have to be the same size; therefore, preprocessing such as size normalisation or resampling may be applied. Generally the unitary transforms are not illumination invariant, except for the Fourier transformed image, as the value at the origin is proportional to the average pixel value of the input image. One main advantage of unitary transforms is that the original character image can be constructed as an inverse transform exists. Examples of unitary transformation techniques which are used to obtain variances of the pixels in transformed space estimation include Karhunen-Loeve transform (KLT), and Fourier, Hadamard and Haar transforms. Some unitary transforms such as Cosine, Sine and Slant transforms are also used. It is found that Cosine transform is comparable to Fourier transform in image data compression [68].

2.4.2 Holes, Vertical Bar, and Cups Feature

Sherkat *et al*. [69] introduced structure feature extraction with the Holes, Vertical Bars, and Cups (HVBC) recogniser. As the name of the recogniser suggests the feature extraction technique employs holistic features namely, vertical bars, holes, and cups; the work was extended from their previous research in on-line cursive handwriting recognition [70]. For vertical bar extraction, the vertical bar is found by calculating pixel density per unit area in each of a word image three zones. The maximum pixel density value in each horizontal position within each zone is likely to be a candidate for the position of vertical bar sections. Bar section in different zones which are close to each other are then joined together to form a vertical bar.

Loop extraction is obtained by tracing the contour of the word image. The starting point is at the right hand side of the direction of travel. When the starting point is met, the tracing is then completed. It is noted that open loops can cause problems as they are similar to the cup feature.The cup feature can be extracted by partitioning the points on external contours into convex, concave and plane regions. One major problem with this feature is to determine if the cup is a basic or non-basic cup. The researchers tried to solve this problem by first finding all cups and then applying a number of heuristics to eliminate the non-basic cups. Example of HVBC Feature of lower- and upper-case character sets can be seen in Figure 2.8.

The discriminations were selected as they were considered to be sufficient to cover all characters. They can be extracted from a word image, then that word image can usually be correctly recognised [71], [72], [9]. It was reported that with a database size of 1,000, recognition rates in the top 50 was 89.28% [69].

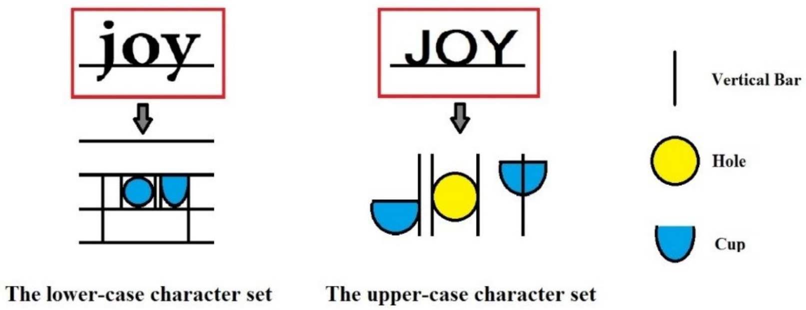


Figure 2.8: HVBC feature of lower- and upper-case character sets [9]

2.4.3 Direction and Transition Feature

Direction Feature (DF) extraction [8], [73], [74] sought to simplify each character's boundary through identification of individual stroke or line segments in the image. The new character representation was broken down into a number of windows of equal size (zoning) whereby the number, length and types of lines present in each window was determined in order to provide a normalised input vector to the neural network classification schemes. Four line segments that would be determined in each character image are: 1) Vertical lines, 2) Horizontal lines, 3) Right diagonal and 4) Left diagonal. Intersection points between each type of line are also located. In order to extract the direction features, first of all the starting point and intersection point of the character have to be located. This can be done by finding the first black pixel in the lower left hand side of the image. The foreground pixels which have more than two foreground pixel neighbours are then marked as intersection points.

Following the starting and intersection points, individual line segments distinguishing is done by following the neighbouring pixels along the thinned pattern/character boundary from the starting point to known intersection points. The algorithm conducted a search in a clockwise direction upon arrival at each subsequent intersection to determine the beginning and the end of individual line segments. After an individual line segment is located, the black pixels along the length of this segment are coded with a direction described earlier. Line type normalisation was also applied so that spurious direction could be discarded. Examples of research using this feature can be found in [75], [76].

Transition feature (TF) is based on the calculation and location transition features from background to foreground pixels in the vertical and horizontal direction. One advantage of this feature extraction technique is that it does not require resizing. Examples of work employing this technique can be found in [77], [78], [79].

2.4.4 Modified Direction Feature

The Modified Direction Feature technique builds upon the Direction Feature Extraction. The main difference in MDF is the way the feature vector is created. For MDF, feature vector creation is based on the calculation of transition features from background to foreground pixels in the vertical and horizontal directions. Both Location Transitions (LTs) calculation and the direction value at that location are stored. The feature extraction processes include 1) Determining LT values, and 2) Determining Direction Transitions (DTs) values. To determine LT values, scanning the image in each row from left to right and right to left is necessary. Each column in the image also must be scanned from top to bottom and bottom to top. A fraction value of the distance traversed across the image is attained by computing the LT values in each direction. A maximum value was defined to be the largest number of transitions that may be recorded in each direction.

DT values can be found after a transition in a particular direction is found. The direction value at the position is stored along with storing an LT value. DT values are obtained by dividing the direction value by a predetermined number. As a result, four vectors would be present for each set of feature values. Another process which has to be performed is re-sampling. Re-sampling of the vector is necessary to ensure that the dimensions are normalised in size. Examples of research that employed MDF include [80], [81], [82]. For other pattern recognition problems which employed the MDF can be found in [83], [84], [85].

2.4.5 Gaussian Grid Feature

The Gaussian Grid Feature (GGF) is a relatively new feature extraction technique [86]. The GGF was developed for the signature verification problems; it employs pattern contours as its input. From the contour representation of a name component image, the contour image is divided into 12 × 12 zones of equal size. By tracing the contours in each block, the 4-direction chain code histogram of each block is created. Every step from a pixel to its adjacent one of the four directions, which is either horizontal, vertical, left diagonal, or right diagonal, are tallied.

A Gaussian smoothing filter ($\sigma = 1.2$) is then applied to each directional 12 × 12 matrix. The value of each element of each matrix obtained in the previous step is then normalised by dividing its value by the maximum value of the four matrices. From the two-matrix pairs, horizontal (H) and vertical (V) matrices, left-diagonal (L) and right-diagonal (R) matrices, two new matrices $\oplus$ and $\otimes$ are established by manipulating pairs of matrices of perpendicular directions. The feature vector is formed by merging the six matrices H, V, L, R, $\oplus$ and $\otimes$. The dimension of the output feature vector is $12 \times 12 \times 6 = 864$. By employing this technique, the lowest average error rate value of 13.93% is obtained.

2.4.6 Background and Foreground Feature

Binarised images are in black and white form. Whereas the white is the background, the black is the foreground (character or word itself). Most of the research used foreground information for feature extractions [87] while some used both background and foreground information. Examples of research which used both foreground and background information can be found in [8], [88], [68], [89], [90]. Features such as zoning, centre of gravity, and moments are examples of features extracted from foreground information. Features, such as transition between background to foreground and foreground pixels and water reservoir concepts are examples of the use of foreground and background information. Centre of gravity and zoning features are described as follows:

- ***Centre of Gravity/Mass Feature***: centre of gravity (COG), which is also known by the name of centre of mass or centroid, is the centre to an object's weight distribution. Many researchers have employed COG in both off-line and on-line handwriting recognition work [91], [92], [53], [93]. The COG is used in slant and skew normalisation processes [94], and also used in signature verification systems [95], [82].
- ***Zoning*:** zoning is one of the important statistical features used for character and word representation. A word matrix or character is divided into several overlapping or non-overlapping zones of predefined sizes. After the zones are located, features, for example, average pixel density, histogram and sum squared distance are extracted from each of the zones based on the percentage of black pixels present [96]. In the work proposed by Azizi *et al.* [97], the density and variance of median zone of the word together with the

structural features were retained to be used in the Arabic handwritten word recognition work. The best recognition rate of 87.23% was obtained.

2.4.7 Speeded Up Robust Features

Speeded Up Robust Feature (SURF), which is a robust feature extraction method, was proposed by Bay *et al*. [98], [99]; its operations are speeded-up by selecting interest points of an integral image from the salient features of its linear box-space and the use of box filter techniques [100]. The SURF algorithm is divided into three main steps:

First, 'interest points' are selected at distinctive locations in the image; examples of the interest points are blobs, T-junctions, and corners. Under different viewing conditions, the same interest points which were found repeatedly are examples of the most valuable property of an interest point detector. The SURF detector is based on the Hessian matrix; however, it uses a very fundamental approximation [101].

Second, the neighbourhood of every interesting point is represented by a feature vector. This descriptor, besides being distinctive, also needs to be robust to noise, detection errors, and photometric and geometric deformations.

Third, the descriptor vectors are matched between different images which are often based on a distance between the vectors; Mahalanobis and Euclidean distances are examples of distances used. As the dimension of the descriptor has a direct impact on the time this takes, a lower number of dimensions is desirable.

2.4.8 Feature Extraction in Holistic Approaches (HA)

Holistic theories have suggested that lowercase words are easier to read than uppercase words; furthermore, familiar words such as function words are easier to read than unfamiliar words. They have also suggested that lowered recognition performance can be found when word shape is disrupted, with this degradation being more pronounced for words compared to non-words, and familiar words compared to unfamiliar ones [102]. Examples of the perceptual features include ascenders, descenders and word length [103]. Word length and counts of perceptual features such as ascenders and descenders are global features, where global features refer to the word shape aspects that can be easily and reliably measured. It has been found that if the word length is provided, the word presumed accuracy was increased, and further improved when word shape information was made available. Some additional holistic features found in the literature include number and direction of strokes, the direction and orientation of the outer contour of the word, endpoints, crosspoints, and holes [104], [33], [105]. Features used in HA may be divided into 3 main categories which are low, intermediate, and high level [102].

- ***Low level features*:** low level features include structural features such as stroke direction distributions. These features have been successfully applied on machine-printed words [106]. An HA system proposed by Ho *et al.* [107] used a global reference frame which consisted of four reference lines which are the image upper boundary, the top line, the base line, and the image lower boundary to represent the locations of shape features. The stroke direction distribution is used to describe the shape of a word. It captures the spatial distribution of black pixels belonging to strokes of various directions. The other system proposed by Farag [108] using low level features employed the entire word trace using an 8-directional chain code and could therefore be considered as using low-level features.

- ***Intermediate level features***: intermediate level features include edges, end-points, concavities, diagonal and horizontal strokes. This feature level exposes a greater abstraction for the word image. The other features which may be considered as intermediate level structural features include cusps and extensions and local extrema [102]. Vertical extrema features which are arc, loop, and cusp were used in a whole word or phrase recognition. Guillevic and Suen [109] employed 7 types of global features which are ascenders, descenders, loops, an estimate of the word length, vertical, horizontal, and diagonal strokes in order to recognise a bank cheque legal amount. Mathematical morphology operations were used to extract horizontal, vertical and diagonal strokes. The features used by Olivier *et al.* [110] consist of a stroke set of 8 which were extracted from anchor points. The stroke set include upper loop, lower connection, upper connection, short lower stroke, lower overstroke, lower loop, upper overstroke, and short upper stroke. They also employed a grapheme set, which consisted of strokes that were extracted from a single anchor point. The graphemes were constituted of 42 representations. The set is a graphical fragment of cursive handwriting. Unsupervised learning was used to make the grapheme selection. Sharma [111] has used the directional features for horizontal, vertical, right diagonal, and left diagonal which were calculated from the size normalised image in a numeral recognition system.
- ***High level features***: high level features are the perceptual features which can be easily perceived by human vision. Ascenders, descenders, loops and length are examples of these high level features. The feature category is widely employed for holistic recognition of handwritten words. In machine printing, ascender and descender heights are uniform and easy to detect. With handwriting, on the other hand, these features are very varied because of writing styles; therefore, the accurate detection of these features is difficult. Ascenders and descenders can be detected by using reference lines [53], [112] or can be detected directly from a run-length or contour representation [113]. However, finding ascenders and descenders using reference lines can fail if the image contains large skew, uneven writing, curved baseline, and top-heavy images (the word “Falls” for example). Dots and holes can be detected by using connected component analysis or by chain code analysis. Diagonal strokes and arcs can be detected from a skeletonised image [114].

Another important feature in this level is word length. Word length can be measured by counting the times that the script goes across the centre line as the ratio of this number to a statistic representing the number of traverses of the centre line per letter. This technique was first applied with the on-line technique and later also applied with the off-line technique. However, the performance of this technique is not satisfactory when applied with off-line techniques. Word length estimation can also be measured by the number of lower contour minima, number of vertical strokes and the number of possible segmentation points, where segmentation points are ligatures and breaks [102].

T-bar feature is an example of an early use of the high level feature. In 1962, Earnest [115] used lexicon filter to count ascenders and descenders and the presence of absence of a T-bar in a holistic handwriting recognition method. It may be noted that the accuracy of applying each feature detection algorithm depends on the style and tidiness of the writing. Higher level structural features, however, seem to be more suitable for HA [102].

2.4.9 Feature Extraction in HMMs

HMMs are a statistical model to analyse sequences as the result, and are therefore suitable for handwriting recognition tasks. In HMMs base off-line recognition systems, the idea is to transform the word image into a sequence of observations [116]. Sliding window technique has been widely used by researchers [52], [117], [118], [119] in order to extract features in sequence form for HMMs. In order to build the HMM, a quantitative definition of symbols is needed. The definition of each symbol is required as a feature vector. In order to obtain good features, some factors can be considered [67], these include:

- Features should be rotation, translation and size independent.
- Features should be easily computable.
- Features should be chosen in the way that they do not replicate each other; this to ensure efficient utilisation of information content of the feature vector.

Kundu *et al.* [67] used moments, number of loops, number of T-joints, number of X-joints, scaled vertical to horizontal distance, isolated dots, zero crossing, and approximate semi-circle in East, West, North, South direction features for the definition of symbols. Chen *et al.* [48] used 35 features to represent the character symbols in the feature space. The feature set includes moment features, geometrical and topological features, pixel distribution features and reference line features.

When using frames in feature extraction techniques for HMMs, some of the features which could be obtained are total number of foreground pixels, the mean, second order moment, their positions minimum and maximum, the differences between the minimum and maximum values compared to the previous column, transition features, and number of foreground pixels between upper-line and base line, for example [120].

Some researchers [66] have applied feature combination between foreground and background information in handwritten isolated characters and numeral strings recognition. Foreground features consist of both local and global features. Background to foreground pixel transitions and vice versa are used to create foreground feature vectors. Mean direction and corresponding variance are obtained by means of statistical estimation for each transition. Black pixel vertical projection is used to obtain global features. Background features are based on concavity information in which each concavity feature represents the number of white pixels that belong to a specific concavity configuration.

Concavity features are also used for HMMs classifiers. Some examples of papers which have applied concavity features are [117], [66]. In [117], the researchers retrieved concavity features by extracting local concavity information and stroke direction within a frame.
Gradient-based features are also in used for HMMs. A gradient-based Scale Invariant Feature (SIFT) descriptor which was proposed by Lowe [121] is an example of the gradient-based features; Lowe's technique was later extended to a bag-of-features word spotting method by Rothacker *et. al* [122]. They used bag-of-features representations for estimating a semi-continuous HMM for Arabic handwriting recognition.

2.4.10 Water Reservoir Feature (WRF)

Water reservoir (WR) concept was developed by Pal *et al.* [88]. Initially the scheme was based on features obtained from the concept of water overflow from the reservoir as well as topological and statistical features of the numerals. Later on, the technique was proposed for numeral and off-line handwritten scripts segmentation [123], [124], [125]. WR could indicate the location of touching points and hence it has been used here for character segmentation. Roy *et al*. [90]employed header line and water reservoir concept based features to compute the busy-zone of the word. Examples of top and bottom reservoirs extracted from the word "technique" can be seen in Figure 2.9 (a) and (b). The water reservoir properties are illustrated in Figure 2.10.

Figure 2.9: (a) Original image and (b) top and bottom water reservoirs

The WR principle supposes that if water is poured from the side of a script component, the cavity regions of the background portion of the component where water will be stored are considered as reservoirs of the component. The important properties of each WR are width, height, areas, its size and shape, centre of gravities, relative positions of the WR, base line , water flow level, mid-point of water flow surface and the position of reservoir with respect to bounding box of the touching pattern; the other important feature is the number of WRs [123], [124], [89].

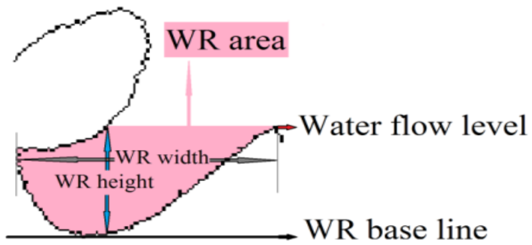


Figure 2.10: Water reservoir properties

2.5 Handwriting Recognition

Handwriting can generally be divided into 3 fundamental categories which are cursive, hand printed (discrete), and mixed style [126] (see Figure 2.11). Each of these styles is classified into 3 further secondary categories which are uppercase, lowercase, and mixed-cased (see Figure 8.). These two classifications, which are style and case, are independent of each other. Handwritten words have both a style and a case [9].

Figure 2.11: Examples of handwriting styles and examples of character cases

Tappert *et al.* [16] classified the problem of handwritten word recognition further into 5 stages, which are:

- Stage 1: Boxed discrete characters where characters are separated by writer.
- Stage 2: Spaced discrete characters where character segmentation is required.
- Stage 3: Run-on discretely written characters where advance segmentation technique is required.
- Stage 4: Pure cursive script writing
- Stage 5: Mixed cursive, discrete, and run-on discrete

Stage 1 can be regarded as a special stage [119]; examples of this stage could be found on examination papers in the name or student ID field, or an envelope postcode field. Stages 1-3 which are hand-printed character recognition can be regarded as simpler recognition problems compared to stages 4 and 5. The reasons are due to the lack or near absence of segmentation problems and fewer variations at the character level. Stages 4 and 5 have been extensively researched. Solutions to the problem include segmentation, whole word (heuristic) recognition, and compromising between segmentation and heuristic recognition. The third solution tries a loose segmentation to find a number of potential segmentation points in the pre-segmentation process and then carries out the final segmentation and the word length determination by using lexica. Example of types of English writing according to Tappert can be seen in Figure 2.12.

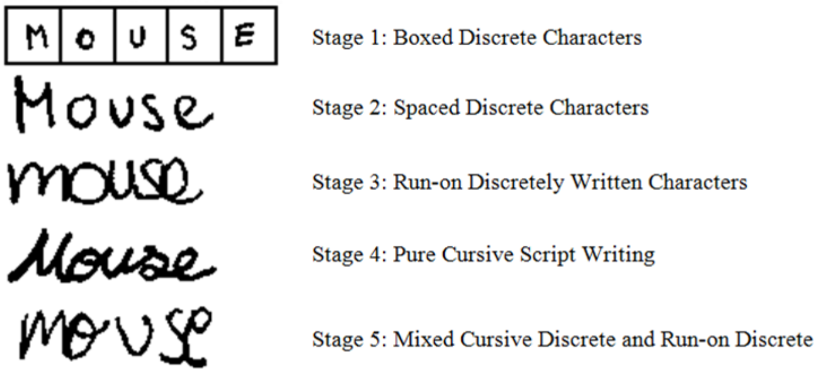


Figure 2.12: Types of English writing by Tappert [16]

Dehkordi *et al* [127] stated that handwriting style characterisations are challenging. This is mostly because of vast variability in handwriting both between different writers (inter-writer) and within the same writer (intra-writer). Furthermore, the variations of handwriting styles may be caused by geographical location, cultural background, sex, and age, for example. Regarding the large handwriting variation styles, it is more likely that style-specific classifiers will result in higher classification accuracy than the generalised classifier. There are two main types of handwriting which are cursive and hand-printed.

a) Cursive Handwriting: Cursive handwriting exhibits a natural variation where the character shape depends on the neighbouring characters and the ligatures between the characters. A character shape is context-dependent, and the shape variation is mainly caused by the ligature connecting between 2 characters. Handwriting properties also include delayed stroke or a diacritical stroke, for example, ‘i’ dot [128]. Cho *et al*. [129] stated that developing the OCR is in fact a difficult task. There are many difficulties existing in cursive handwriting character recognising development which include:

- Large Shape Variation: This difficulty is caused by large variation on a character which is either written by the same writer (intra-writer variation), or by a different person (inter-writer variation).
- Restricted Information: This difficulty is caused by the lack of temporal information in an off-line recognition system, for example sequence and speed, which indeed is important information.
- Quality of the available information: This difficulty is caused by noise or heavily degraded image. Furthermore, various background colour or texture can cause confusion; therefore, the need of a robust preprocessing algorithm is essential.
- Complexity: As cursive handwritten is more complex than discrete or run on handwritten words, complex processes such as segmentation and language model may need to be used to develop a system.

Crucial components of a cursive handwriting recognition system are efficient preprocessing operations, a robust feature extraction, particularly with regard to a writer independent system, and the modelling approaches as well as the usage of contextual knowledge [54]. The off-line cursive script recognition can be classified depending on two key properties which are size and nature of the lexicon involved, and whether or not a segmentation stage is present [52].

It may be noted that an important term used when dealing with cursive handwriting is ligature. Ligature is the transition between two characters. The experiment performed by Dolfing [128] results shows that the ligature shapes have an effect on the recognition performance. If the ligatures are trained together with the character models, the ligatures add an extra source of variation to the character shapes. The recognition process depends on the character shapes; therefore, [128] tried to minimise the effect of the ligatures on the character shapes with either contextual or ligature models.

b) Hand and Machine Printed: In handwritten words, the characters are not explicitly joined through ligatures. However, the characters may be written very close together so that slant and slope can be found. Slanted characters can cause an overlapping problem [8], and as the characters are written so close together the touching components may present.

Machine-printed words are similar to hand-printed as the characters are not joined through ligatures. However, the characters can be stained by hammer tilt, ink splatter, and other imperfections in the mechanical structure of the system [130].

Handwritten characters are very different from the printed character as the writing's local features have a non-linear distortion and variation. These include the problem of writing environment such as inequality of image thickness [65]. Plamondon *et al.* [2] stated that recognition accuracy of a single machine-printed font family character on a well-printed document can be very high. However, it is more difficult when recognising the hand-written characters.

Recognition includes a comparison of the unknown test pattern with each class reference pattern, and measuring a similarity score between the test pattern and each reference pattern. After the scores are calculated, the pattern comparison scores are used to determine which reference pattern best matches the unknown pattern. There are many recognition techniques available such as Hidden Markov Models, Dynamic Programming, Neural Networks, K-Nearest Neighbour, expert systems, and hybrid. Other sources of knowledge, such as language models, can also be used in the recognition process to improve the recognition results. The lexicon is the most commonly used as it represents the recognition vocabulary that is expected [22].

As stated by Koerich *et al*. [22] lexicon is a list of all valid words that are expected to be recognised by the system; therefore, it limits the number of possible word hypotheses to be searched. It is important for the recognition process as it is the source of linguistic knowledge. Linguistic knowledge disambiguates single characters by looking at the entire context. The recognition systems generally rely on the lexicon during the recognition process. This is called a lexicon driven system. The lexicon can also be used after the recognition as a post processor of the recognition hypotheses. Contextual knowledge including linguistic, domain, or other pertinent information is required to be incorporated into the recognition process to reduce the ambiguity and attain acceptable performance. Nevertheless, the systems that rely on a lexicon in the early stages have had more success as they look directly for a valid word [22].

The number of words in the lexicon is also important. The greater in size, the greater computational complexity becomes as there are more similar words which are more likely to be present in the lexicon. A limited vocabulary can be the key aspects of a system that relies on large vocabularies, as it contributes to improving the accuracy as well as reducing computation. Some additional techniques can be included to improve the recognition results including pruning or lexicon reduction mechanisms. Lexicon size depends on the application environment; furthermore, lexicon sizes are usually categorised as:

- Small vocabulary which is tens of words e.g. legal amounts on bank cheques
- Medium vocabulary which is hundreds of words
- Large vocabulary which is thousands of words e.g. postal application
- Very large vocabulary which is tens of thousands of words

With any large vocabulary recognition systems, there are two issues to be considered which are accuracy and speed. Generally the problem with accuracy is common to small and medium vocabularies, even though the task of recognising the small to medium vocabulary words is much easier than the one from a large lexicon. With the large lexicon recognition system, speed is an issue as the computational complexity increases because of the presence of more similar words in the vocabulary.

One main problem with large lexicons is the number of times that the observation sequence extracted from the input image has to be matched against the words in the lexicon. In order to reduce the number of words to be compared during the recognition, some techniques such as pruning methods are applied. However, one issue that needs to be considered is that there is a chance that the true word hypothesis may be thrown away. Some of the basic ways to achieve a lexicon task include the knowledge of application environment, performance of a pre-classification of the lexicon entries to evaluate how likely the matching with the input image will be by looking at word length and word shape, the use of sources of knowledge such as contextual knowledge, and the use of language syntax.

2.5.1 Numeral Recognition

One of the well-known problems in pattern classification is off-line handwritten numeral recognition. The recognition of handwritten numerals plays an important role in OCR research and development because of its many potential applications such as bank cheque processing, postal mail sorting, automatic reading of tax forms and various handwritten forms. The challenge is the variety of handwriting styles, shape, and size. Multi-Layer Perceptron (MLP) trained with back propagation algorithm provides an efficient classifier for handwritten numeral recognition [111].

2.5.2 Word Recognition

Word recognition algorithms can be divided into two main groups which are an analytic approach and holistic approach [131]. The details of both groups are discussed below.

2.5.2.1 Analytic Approach / Segmented / Sub-word Recognition

Analytical or segmentation-based approaches determine the identity of the word from the identities of smaller units such as character [103]. These analytical strategies employ bottom-up approaches, starting from stroke or character level and moving towards producing a meaningful text. Segmentation, either explicit or implicit, is required to segment the word into characters or strokes. By applying segmentation, the recognition problem is reduced to recognition of simple isolated characters or strokes. Therefore, the other advantage of these approaches is that they can handle unlimited vocabulary [3]. For these reasons, the analytic approaches are more suitable for large vocabulary applications [53].

In these approaches, the word is segmented or is broken up into characters or sub-word units. The segmentation consists of isolating the single characters in a word so that each one of them can be separately recognised. Such a task is difficult and error prone since a character cannot be recognised before having been segmented, but cannot be segmented before having been recognised. This is referred to as the Sayre’s paradox [52].

A character recogniser is used to recognise many hypothesis characters made up of combination of the smaller over segmented slices. To be able to recognise these sub-words, the recognition strategies used are fundamentally based on dynamic programming methods, which try to match primitives or blocks of primitives with sub word units to recognise word. With the optimal path problem, DP methods are used find the optimal sequence of a fixed number of moves. The word recogniser combines the most suitable combination of hypotheses that gives the highest word score among all words in the lexicon. The best word score eventually determines the ultimate segmentation points of the word [17]. The analytical approaches which are used in large vocabulary handwriting recognition can divide into two categories [22] which are:

- **Character recognition followed by word decoding–characters or pseudo-characters is the basic recognition units.** They are modelled and classified independently of the words. In this category, pattern recognition approaches such as template matching, structural techniques, neural networks and statistical techniques can be used. Character scores are also matched with lexicon entries by DP methods.

- **Character recognition integrated with word-decoding characters or pseudo-character are the basic recognition units.** The basic recognition units are concatenated to build up word models according to the lexicon. DP methods are also used to evaluate the best match between the word sequence of observations and the word models.

2.5.2.2 Holistic Approach (HA) / Word-based Approach

There is evidence from psychological studies of reading that points out that humans use perceptual features or global shape in addition to letter identities in fluent reading [132], [133]. There are quite a few terms defining word shape depending on holistic theories; these include word envelopes, shapes and sizes of individual letters, arrangement of ascenders, descenders and neutrals diagrams and spelling units.

Holistic strategies employ top-down approaches for recognising the whole word [3]. In handwriting recognition, algorithms based on word-shape features are often called holistic. The holistic or word-based approach is to treat and recognise the word as a whole based on features and feature sequences [18], [102]. Each word is generally modelled individually [17]. This approach, to some extent, avoids problems caused by the broad variability of character shapes, segmentation ambiguity in segmentation approach, and premature character recognition [107]. Furthermore, the computation of holistic features is inexpensive [23]. In HA, since every word is treated as a different class, features and matching scheme used have to be coarse enough to be stable across exemplars of the same word class. As words can be considered complex 2D patterns, and the writing styles can be large, it is difficult to satisfy these criteria when the class number is large or unknown. Since the lexicon is small and fixed, the larger amounts of training samples for each word class can be collected. HA is not suitable for large or dynamic lexicon because the separability of word classes in feature space is considerably reduced and overwhelmed by intra-class variations. For the reasons described, HA has been used traditionally in applications where the classes are small and static [102].

The majority of applications based on HA involve a small, static lexicon of possible words. As mentioned, the lexica used in HA is relatively small at around 10-100 words. With this approach, the larger the lexicon size is, the lower recognition rates become. general weakness of holistic word recognition is a large lexicon is needed that holds every possible word that could be written , therefore there is no possible way to recognise word images that ware not in the lexicon (compare to character based). With large or dynamic lexicon, the used of lexicon reduction and verification are generally applied. Verification is the task of verifying that a given image is that of the given American Standard Code for Information Interchange (ASCII) string. Examples of the applications which use a holistic approach include interpretation of handwritten postal addresses, handwritten responses on forms recognition, and especially cheque amount recognition tasks [104].

2.6 Applications

There are many applications that benefit from off-line handwriting recognition techniques. Most well-known applications are the signature verification system, postal address interpretation, and cheque verification. These applications are practical, and in use worldwide.

2.6.1 Writer Identification and Verification, and Signature Recognition and Verification Applications

Writer identification is the task of determining the author of a sample handwrtiting from a set of writers. Writer verification is the following task of deciding whether or not a handwritten text has been written by a certain person. To use any text to establish the identity of the writer, is considered a text dependent identification task. On the other hand, a text dependent task involves a writer in writing a particular predefined text to identify themselves [134]. Signature identification and verification application is one of the applications that is concerned with the biometric human identification field which benefits from both on-line and off-line handwriting recognition techniques. Signature verification is also one of the convenient human identification methods; it is the task of determining whether a particular signature belongs to a specific writer, and signature recognition is the task of deciding to which of a certain number of writers that particular signature belongs to [135]. Off-line signature verification is still being explored. Researchers [136] also applied Modified Direction Feature and the Radial Basis function neural networks in their work; the verification rate is as high as 91.12%.

2.6.2 Postal Address Interpretation

Postal address interpretation is now in use worldwide, and it is also one of the most successful applications which benefits from an off-line handwriting recognition technique. In 1965, the United States Postal Service began reading the city/state/ZIP line of printed envelopes using OCRs [137]. By 2009, in the USA, reading rate of handwritten mail pieces was around 95%, with an error rate of less than 3% [127]. The main task of the system is to determine the destination address on envelopes [138]. There are problems in the recognition process as the images of destination addresses are often of poorly written, incomplete or incorrect. In order to recognise the address, the segmentation and the neural network recognition techniques have been used [32].

2.6.3 Cheque Verification

Cheque verification/bank cheque reader is one of the most successful handwriting recognition applications [80]. Handwriting recognition has been used on a cheque, such as date, signature verification and amount of money. The amount of money on a cheque includes both the courtesy amount and legal amount. To verify the mentioned features, many processes such as segmentation and recognition will be applied [139]. Feature extraction and neural networks are used in the recognition process.

2.6.4 Word Spotting

Word spotting concerns the process of retrieving all instances of a given keyword from a document in any language. Great interest has been taken on handwritten word spotting in different application areas: an important application for modern handwriting is automatic mail sorting. There are two approaches in handwritten word spotting found in the literature which are template-based and learning-based methods [140]. Frinken et al. [131] proposed a keyword spotting method for handwritten documents by deriving them from a neural network-based system for unconstrained handwriting recognition. It performs template-free spotting, for example; it is not required for a keyword to appear in the training set. A Chinese keyword spotting method was also proposed [141]. On a text query word, the measurement of the similarity between the query word and every candidate word in the document took place; this was done by combining a character classifier and four classifiers characterising the geometric contexts. The retrieval rate of 86.43% was obtained.

2.6.5 Automatic Assessment Systems

Automatic assessment systems are the other application that benefits from handwriting recognition. They can be divided into 2 categories which are on-line and off-line assessment systems. The on-line assessment system is practical and is being used worldwide. For off-line assessment systems, a well-known technique is Optical Mark Recognition (OMR) which has also been employed in other applications such as surveys, questionnaires and membership subscription forms. It should be noted that there is still no recent further development in off-line automatic assessment systems. There are only a few existing research studies with regards to off-line automatic assessment systems [9], [10], [11], [12], [142], and limited work has been done for automated essay grading [14], [13].

2.6.5.1 Automatic Assessments / Examinations Overview

In order to be able to develop the proposed system (automatic system for assessing off-line handwritten short answer exam questions), knowledge and understanding of examinations and their assessment methods are required. Examinations are very important within the education system, as they help teachers to measure their students' knowledge, skills, and development, and thus enable teachers to plan for their students in the future. One important factor to consider when producing examinations is that they have to be valid. Validity of the examinations is determined by answering these questions:

- Do the examinations meet the learning objectives of the subject?
- Do they evaluate what teachers would like to evaluate?

- Do they evaluate the desirable outcomes of the subjects?

Important features of an examination which include examination types, seen examinations, unseen examinations [15], and the existing automatic off-line assessment systems are described as follows:

a) Examination Types: examinations can be generally divided into 2 types, which are formative and summative. Both of them have unique characteristics and different usability; therefore, it is important to understand and be able to decide which type is to be used in each particular assessment.

1) Summative assessment, as the name implies, is used to evaluate students' sum of knowledge after a period of time of studying. Hence, the summative assessment is used for recording the overall achievement of each student in a systematic way. Summative assessment can also be used for the purpose of reporting the students' achievements to parents, the other teachers and most importantly, the students themselves [143], [144]. Summative assessment can be divided into two categories which are seen and unseen examinations [9].

Unseen examinations are traditionally summative assessments in which the students do not know what questions will be presented in the examination. The summative assessment also has many forms of questions, for example short answer questions, essay, multiple choice questions or projects. The students do not have any aids or resources to complete the examinations [9].

Seen examinations, as the name indicates, gives the opportunity for students to prepare themselves for the examination better as they know what to expect. The students can prepare themselves before the examinations by accessing the resources such as textbook or other materials. Also for this type of examination, the students may also be allowed to use text books or other materials in the examinations [9].

2) Formative Assessment is used to learn about students' existing knowledge, skills and ideas. It also has to be carried out by teachers. Formative assessment is used to decide if a student has achieved an adequate level of skill of some subject content and if they are ready to take the next steps. In other words, formative assessment is used so that the positive achievements of a student may be recognised and discussed and the appropriate future can be planned. This type of assessment is essentially feedback to both the teacher and the student. For the students, it helps them to realise their understanding and their skills development. For the teacher, it helps them to learn about the students' skills and understanding, and therefore helps them to decide the future plan for the students' learning [143], [144].

Multiple Choice Examinations (MCEs) have the simplest style of questions. Multiple choice questions consist of a prompt which can be a question, or the start or end of a sentence which the student must then complete by selecting the correct answer from the given a number of alternative answers. MCEs became popular in the beginning of the last century. The reason for this was that researchers became more aware of the essay examinations' limitations. MCEs have many important features. First of all, they are efficient and ambiguity free [145] as the multiple choice examinations require the students to select their answer from among a number of alternatives. Students are not required to write down their own words; only the ability to select the best response from the alternatives is required [66]. MCEs are therefore easy to score and do not require a lot of time to mark.

Even though MCEs are easy to mark, the time required to produce one can be hours. The other features of this type of examination are that the preparation time, marking and modification is not as dependent on the amount of students doing the test [146]. MCEs are well-respected and most of the United States use the format to conduct the majority of their tests, such as the Scholastic Aptitude Test (SAT), the American College Test (ACT), the Graduate Management Admission Test (GMAT) [146].

Short Answer Examination (SAE) and Essay Examination (EE) questions are constructed-response questions. The answer to Short Answer (SA) questions can be in the form of fill-in-the-blank, or in the form of sentence(s) [146]. No alternative responses are provided for the students to choose from. For SA examinations, the answer from each student can vary. As it requires the creative capacities of individuals, the student can express their answer in their own way depending on their understanding of the questions. SA questions have similar characteristics to the EEs. The differences are that the answers to the SA could be less creative, as the questions themselves require definite answers, and the answers are usually shorter than the essay examinations [66]. It must be noted that marking these types of examinations is more time-consuming compared to the MCEs [146]. Generally, when marking the SAE/EEs, full or partial marks can be given depending on the completion of the answers and if they are correct.

The choice of MCEs and SAEs/EEs is controversial. Many researchers believe that SAEs/EEs promote students' learning and logical thinking better than MCEs examinations, as they are more difficult than the MCEs and therefore require a greater application of knowledge. They also believe that in order to be able to do SAEs/EEs, the students need to recall whatever they have learnt as opposed to the MCEs, where the students only need to recall and recognise the best response from the alternatives [67].

Some researchers also believe that SAE/EE examinations require more creativity than the MCEs, as students have a chance to use self-expression to do the examinations [66]. Some researchers also believe that MCEs are unsuitable to evaluate or to promote students' critical thinking. Paradoxically, it has been found that most popular critical thinking tests are in MCE format despite their believed unsuitability [146].

It has been reported that the MCE format is as valid and reliable as the SAEs/EEs with the only difference between the two being that the SAEs/EEs take the same or more time to complete. It was also found that the MCE format may lead higher reliability. However, because this research was performed by using questions mainly in the lower level of Bloom's taxonomy, the result may have been different than if the questions had been in the higher level [146].

In conclusion, SAEs/EEs and MCEs have their own advantages and disadvantages. One important aspect is that the teacher carefully selects an assessment method that is suitable for the subject and its content, as it will help to promote the students' skills and their knowledge [147]. In addition, it is important to consider that it does not matter how well the examination format is selected if the questions themselves are not valid.

b) Problem with Assessments: the time used to produce assessment can be considered long. However, the time spent marking the examinations is much longer, especially formal ones that might needed to be moderated [9]. Additionally, the inconsistency in marking each student's examination answer may occur even though the students' answers are very similar. The marking process, especially handwritten SAEs/EEs, is difficult. It requires the markers' concentration, precision and it is very time-consuming. As examination assessment is very important, the marking process needs to be done properly and correctly as the price of marking the examination wrongly is too high.

c) Computer Assisted Assessment (CAA): CAA is a general term for the use of computers in student learning assessments. In this case the computers are used to deliver, mark and analyse examinations or assignments. It also includes the collation and analysis of optically captured data gathered from machines such as optical mark readers (OMR). Other CAA activities include computer-based testing, computerised assessment, computer-aided assessment and web-based assessment. Computer-based assessment (CBA) is also another important term to describe an assessment in which the questions or tasks are delivered to a student through computer terminals [148]. There are many advantages to use CAA which include:

- As the frequency of assessment increases, the students have more motivation to learn and are encouraged to practise skills.
- It broadens the range of knowledge assessed.
- It increases feedback to both students and teachers.
- It extends the range of assessment methods.
- It decreases marking time and loads.
- It aids administrative efficiency.

Some of the automatic assessment technologies are described as follows:

1) Optical Mark Recognition: Since the 1950's, Optical Mark Recognition (OMR) has been used for Marking multiple Choice Questions (MCQs). The OMR is said to be a fast and accurate MCQ marking tool [149]. OMR is a machine which rapidly processes paper forms by scanning the page for marks such as shaded boxes, crosses or ticks. Applications that use OMR technology include, for example the National Lottery, student questionnaire processor, and MCQ marker [148]. To use a MCQ marker, students use a pen or pencil to mark on a pre-printed paper form to indicate each selected response [149]. The completed forms are then scanned afterwards by an OMR that senses the presence of a mark by measuring the reflected light (see Figure 2.13).

2) Automated Essay Grading System (AEG): Off-line AEG was invented to mark off-line essay examination [14], [13]. An example of an essay answer sheet can be seen in Figure 2.14. The approaches applied to the system are based on document image analysis and recognition together with automated essay scoring. The system [13] deals with children's handwriting style, and the recognition is based on a fusion of children's holistic and analytic methods combined with contextual processing based on trigrams. The examination was constrained by a reading passage, question and rubric. The lexicons to recognise handwritten words are derived from the aforementioned materials. Scoring is based on two essay scoring methods, namely 1) Latent Sematic Analysis (LSA) which requires a reasonable level of handwriting recognition performance and 2) the uses of ANN which is based on features extracted from the handwriting image. Whereas the LSA requires the use of a large lexicon for recognizing the entire response, ANN only requires a small lexicon to populate its features thereby making it practical with current word recognition technology. The researchers stated that the evaluation of the system was not so much on recognition rates but in terms of the overall application in which it was used.

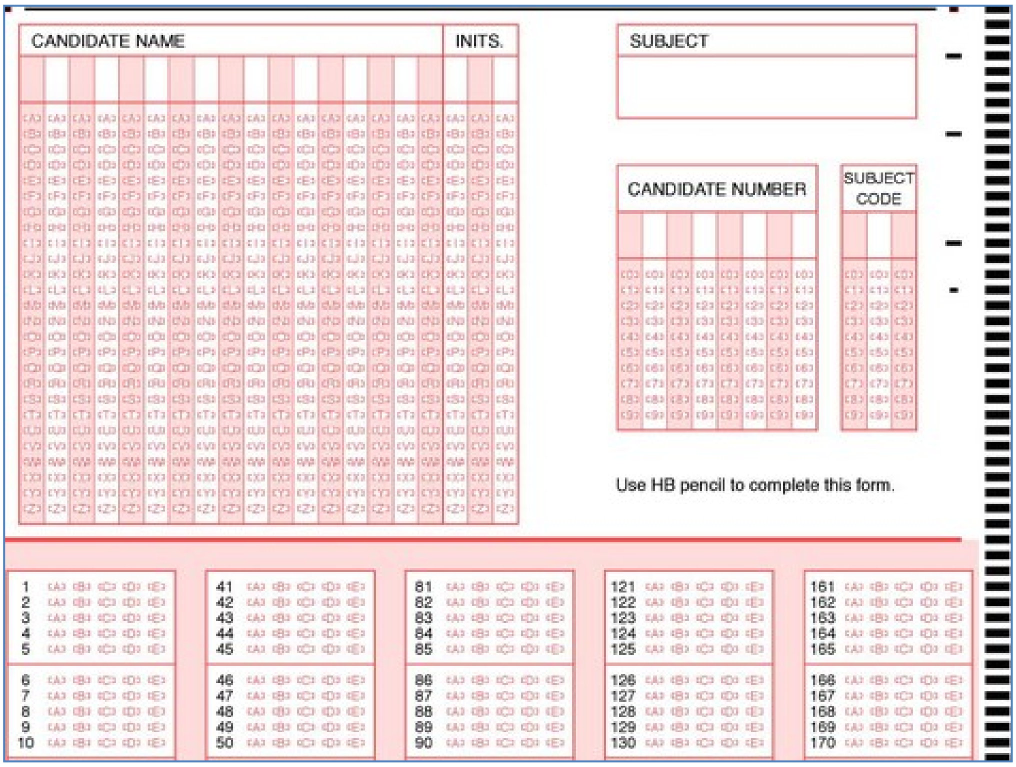
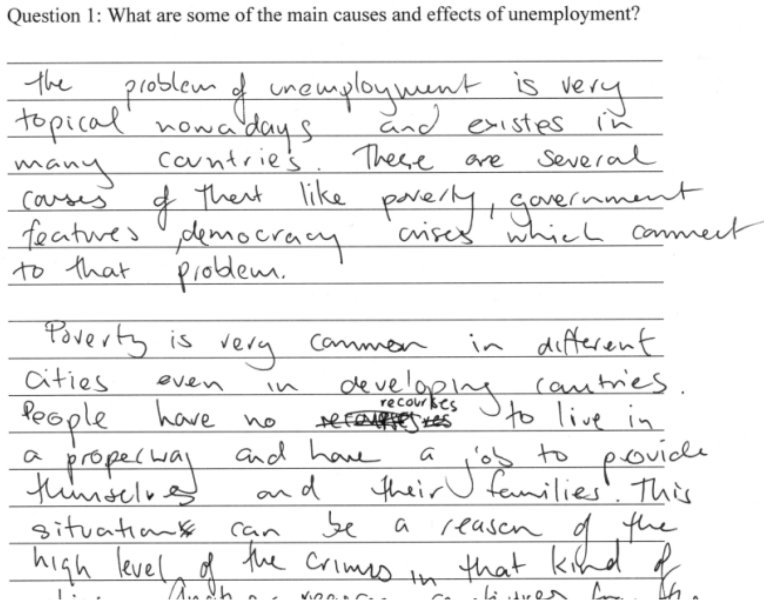


Figure 2.13: An example of an OMR answer sheet [150]


Question 1: What are some of the main causes and effects of unemployment?

the problem of unemployment is very topical nowadays and existes in many countries. These are several causes of that like poverty, government features democracy crises which connect to that problem.

Poverty is very common in different cities even in developing countries. People have no recourses to live in a proper way and have a job to provide themselves and their families. This situation can be a reason of the high level of the crimes in that kind of

Figure 2.14: An example of a cause and effect short answer response

i. **On-Line Automatic Assessment Systems (OLAASs):** Recent trends towards teaching large numbers of students at low cost have led many researchers to explore ways in which the assessment of student assignments could be automated [151]. In addition, there is much research on the impact of computer-based formative and summative assessment on student learning and performance that has become increasingly important [152]. This reflects wide use of OLAASs [151]. It was found that within the USA, there are an increasing number of states which have adopted Automated Essay Scoring (AES) programs in school and classroom based writing assessment as well as in state summative writing assessment [151].

One example of an automated SA/EE assessment system using Natural Language Processing (NLP) was proposed by Burstein *et al*. [153]. The scoring rate achieved was 97%. This scoring rate was calculated based on the number of automatically assessed responses that are in agreement with a scoring given by a human assessor [9].

It may be noted that even though the OLAASs are used worldwide, there are some limitations in using them. The main problem is that if the students are unable to relate their thoughts efficiently through a keyboard, or the cost of testing done on a computer is expensive, then a complete move from traditional pen and paper based assessment to on-line based assessment is impractical [9].

ii. **Off-Line Automatic Assessment Systems (OFLAASs):** Traditional pen and paper based assessment are still the most common form of examination at every level of education and it is a time-consuming process marking these examinations paper. Success in developing OFLAASs could relieve the work load and marking time of assessment for teachers.

The main problem in developing such systems is that the marking confidence and accuracy would have to be extremely high. The price of marking the examination incorrectly, which may be caused by the systems misrecognised answers or marking them wrongly, is too high.

To the writer's knowledge, existing OFLAASs have only been developed by Allan *et al.* [9], [10], [11], [12]. Allan *et al.* had proposed a system which assesses the examination responses from whole words to sentences. This work achieved the assessment yield range from 54% with 99% accuracy to 33.2% with 100% accuracy depending on the constraints used, such as lexicon and bridges between the lexicons, and the response history applied. In such systems, a structural feature extraction technique with a holistic HVBC recogniser has been used.

Although the recently developed systems in this area have achieved an encouraging assessment yield range executed under both constrained and unconstrained conditions, there is still not much work in the development of off-line automatic assessment systems using handwriting recognition, even though such systems will clearly benefit the education sector.

2.7 Classification

Classification or recognition includes a comparison of the unknown test pattern with each class reference pattern, and measuring a similarity score between the test pattern and each reference pattern. After the scores are computed, the pattern similarity scores are used to decide which reference pattern best matches the unknown pattern. There are many recognition techniques available such as HMMs, Dynamic Programming (DP), Artificial Neural Networks (ANNs), K-Nearest Neighbour (KNN), Support Vector Machines (SVMs), expert systems, and hybrid. As a classifier has to be trained using the available training samples and their values, the criteria of choosing a classifier depends heavily on the nature of the samples' features used [154]. The other source of knowledge such as language models can be used in the recognition process to improve the recognition results. The lexicon is the most commonly used as it represents the recognition vocabulary that are expected.

2.7.1 Artificial Neural Networks (ANNs)

Artificial neural networks are inspired by biological neural networks. In 1943, Warren McCulloch and Walter Pitts [96] introduced models of neurological networks based on mathematics and algorithms named threshold logic, and showed that simple networks are able to calculate nearly any logic or arithmetic function. A set of inputs are applied, with each input representing an output of another neuron. Each input is multiplied by the corresponding weight. The weight inputs are then summed, similar to the biological neuron, to determine the activation level of the neuron. The activation functions, which may be linear or non-linear, are used to produce the neuron's output signal.

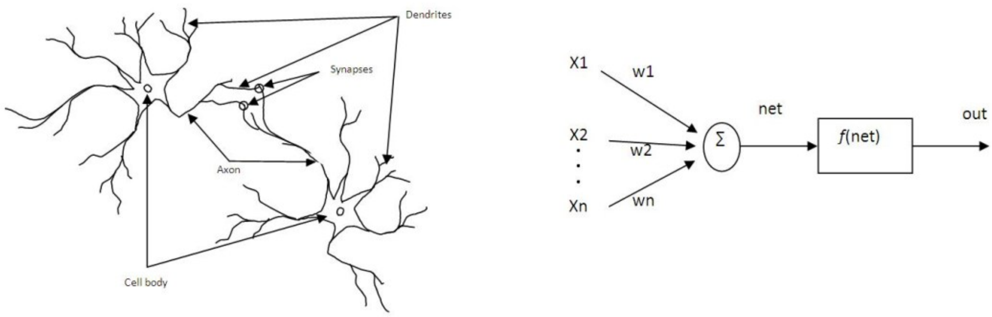


Figure 2.15: Biological neurons and an artificial neuron

The neural network has been widely used for classification problems. The main advantages of the neural network lie in its ability to be trained automatically from examples of its good performance with noisy data, and in its being an efficient tool for learning large databases. It has been successfully implemented in feature selection, object tracking, biometrics, document and image preprocessing, analysis, and classification [25].

There are single- and multi-layer perceptrons. A Single-Layer Perceptron (SLP) has one layer of variable weights, and one layer of output neurons. A classic boolean funtion AND or OR can easily separated/classified by employing SLP as it is a two-class problem. However, employing SLP can be limited as it can only represent linearly separable data (see Figure 2.16). The Multi–Layer Perceptron (MLP) is a feed-forward type neural network; as the name suggests, the information flows from the input toward the output. The MLP has two or more trainable weight layers, and as a result, it is able to solve non-linear problems. For that reason, the most widely used and studied neural network is MLP [111].

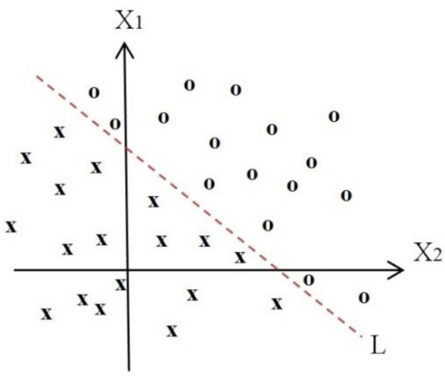


Figure 2.16: Linearly Separable Pattern

To use neural networks as classifiers, they need to be trained on a set of data. The training algorithm adjusts the weights between each connection until some criteria are met [8]. Neural networks, especially multi-layer neural networks, have been used in handwriting recognition problems for decades. This is because MLPs yield exceptional performance in pattern classification tasks for which the input is known to be from a finite set of pattern classes.

However, the drawback of MLPs is that they can also yield high outputs when a sample that is not from one of the classes is presented as input [155]. An example of a feedforward network with three layers: two input neurons, three hidden neurons and three output neurons (Ω) can be seen in Figure 2.17.

Koerich *et al*. [22] stated that handwriting recognition using neural networks has mostly been applied on digit recognition, isolated character recognition, and small vocabulary word recognition. The reason that neural networks are not generally used in large vocabulary handwriting recognition is because words must be segmented before neural network modelling. However, they can be used with large vocabularies as part of hybrid approaches, where they are used as a back-end classifier to estimate a priori class probabilities, a priori grapheme probabilities or to verify results of previous classifiers. With large vocabulary, however, neural networks are not frequently used as front-end classifiers.

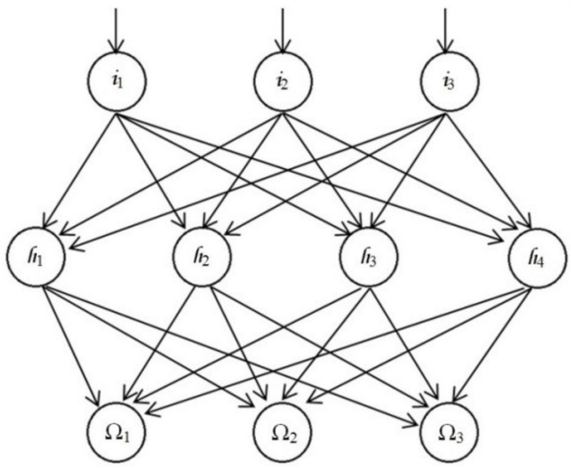


Figure 2.17: A feedforward network with three layers

A research by Maddouri *et al.* [156] employed a specific neural network called Transparent Neural Network, combining global and local features of the words. A recognition rate of 97% was achieved on a 70 Arabic word lexicon [157]. Mehta *et al.* employed probabilistic neural network in their automatic cheque processing system; the neural networks were used for courtesy digit and payee field recognition. ANNs also used in hybrid classification models. An example of such model is the work proposed by [158]. In their off-line handwriting recognition work, they integrated neural network language models in the decoding process of three state-of-the-art systems namely bidirectional recurrent neural networks, hybrid hidden Markov models and, and a combination of both.

2.7.2 Hidden Markov Models (HMMs)

An HMM is a doubly stochastic process with an underlying stochastic process that is not observable (hidden), and which can only be observed through another set of stochastic processes that produce the sequence of observed symbols [159]. HMMs can be considered as a generalisation of dynamic programming technique. Initially it was used in the field of automatic voice recognition. There are many of applications which benefit from HMMs techniques including speech recognition [137], language modelling [160], handwriting recognition [52], [161], and off-line signature verification [162], [20], and word spotting [140], for example. In the area of handwriting recognition, an important advantage of employing HMMs is that it does not require explicit segmentation of the text into recognition units (e.g. characters or strokes) while most of the other classifiers require some forms of text, word, or character segmentation.

HMMs are a statistical model to analyse sequences and are therefore well-suited for handwritten text recognition. Assuming that a statistical process created the observed sequence, the model is interested in identifying the internal states of the creating process at each time step. These internal states can then be mapped to a sequence of letters which constitute the recognised word [163]. In general, the handwritten data is fragmented into parts that are supposed to belong to some finite set of basic strokes. The vectors extracted from the fragments should form clusters corresponding to the elements of such a set [52]. A sliding window makes the segmentation unnecessary as by sliding the window column by column from left to right, frames of fixed width are extracted from the image. From each frame, a feature vector is extracted and the sequence of observations so obtained is used as the representation of the word. For each entry in the lexicon, a word-HMM is created by concatenating single letter HMMs. The use of letter models makes the system flexible with respect to changes in dictionary, and makes the use of large lexica possible since it does not require training examples of each word. In the case of Arabic, the window moves from right to left. Most major recognition system are based on either Bakis or linear topology [52]; the models topologies can be seen in Figure 2.18.

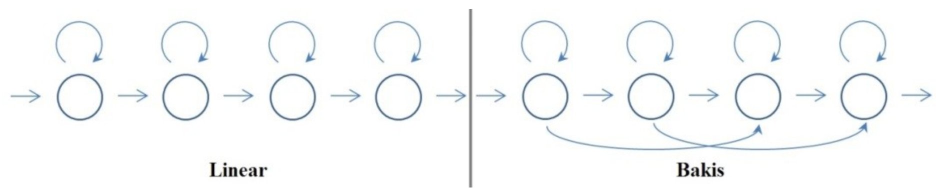


Figure 2.18: Linear (left) and Bakis (right) model topologies used for elementary units in HMM-based writing models [164]

With Markov modelling techniques, explicit alignment as in dynamic programming is not required. A probabilistic transition and observation structure, however, is defined for each written word. This structure is called a Markov model [165], which consists of:

1) A state transition matrix.
2) An observation probability matrix for discrete probability densities or a set of continuous densities defined by parameter sets, or a mixture of the two when different types of densities are used.
3) An initial state probability vector.

which can be represented by a compact notation $\lambda = \{ A, B, \pi \}$ where A is the state transition probability distribution, B is the observation symbol probability distribution in states, and π is the initial states distribution [159]. For each written word v, in a vocabulary of V words, a HMM is formulated from a training set of data representing multiple occurrences of the vocabulary of written words. In general, there are three different modelling techniques used for HMMs which are continuous, discrete or hybrid [54]. Details of these techniques are discussed below.

- **Continuous HMM:** with the continuous HMM, the observations are continuous. A continuous probability density function is usually approximated by a mixture of normal distributions [166]. For continuous HMM, a research paper with no segmentation process by [52] stated that by not having segmentation performed, the observations are vectors extracted from blindly isolated frames (as from a sliding window) and are supposed to be distributed more uniformly than that in discrete HMM. This might result in a large quantisation error because of the high variance within classes. The researchers also added that for this reason, the use of continuous density HMMs has often been referred to. In the work the observations were modelled with mixtures of Gaussians.

Some researchers [48] have used a continuous density variable duration hidden Markov model to recognise unconstrained handwritten words. A mixture of Gaussian distribution is used to model the symbol probability distribution of each state, where a state refers to a character.

- **Discrete HMM:** the HMM is said to be discrete if the observations are naturally discrete or quantised using vector quantisation, and if drawn from an alphabet or a codebook [166]. Feature vectors are mapped to symbols by using vector quantisation. These symbols are referred to as code words while a set of symbols are called codebook [167]. Vinciarelli *et al.* [52] who applied discrete HMM in their work stated that the process that extracts the strokes from the word is called segmentation. At the end of the feature extraction process, the word is reduced to a sequence of symbols that is given as input to a discrete HMM. Discrete HMMs have also been used to combine with a vector quantization step in order to be able to handle continuous feature representations [164].

- **Semi-continuous HMMs:** the semi-continuous is another family of HMMs. It is a compromise between discrete and continuous HMMs which mutually optimise the vector quantised codebook and HMM parameters under a unified probabilistic framework. Ahmad *et al.* [168] proposed sub-character HMM models for Arabic text recognition; some other work which applied semi-continuous HMMs could be found in [169], [170].

- **Hybrid HMM:** hybrid HMMs are invented by combining other classifiers with HMMs in order to improve recognition rate. The work by [54] proposed the discrete, and two hybrid techniques which consist of a discrete and a semi-continuous structure. The other [126] proposed a combination system of HMM and fuzzy logics for handwriting recognition classification. A combination of HMM and ANN could be found in [171].

2.7.3 Support Vector Machines (SVMs)

The SVM which is related to statistical learning theory and is fundamentally a two-class classifier was first introduced by Vapnik [172]. SVMs implement the concept of mapping an input vector into a higher dimensional feature space through a non-linear mapping (see Figure 2.19). It is a training algorithm of learning classification and regression rules from data. It can be employed to learn polynomial, radial basis function and MLP classifiers. Two crucial components in SVM implementation are the techniques of mathematical programming and kernel functions. Because kernel functions are flexible, it allows the SVM to search an extensive variety of hypothesis spaces. Additionally as it uses statistical learning theory to search for a regularised hypothesis that fits the available data well without over-fitting [173].

From Figure 2.20, it can be seen that the SVMs in their simple form are hyperplanes which separate the training data by an optimal hyperplane which is the one with maximal margin (d+, d-); d+ is the shortest distance to the closest positive point and d- is the shortest distance to the closest negative point. Therefore d+ plus d- is equal to the margin of a separating hyperplane. The support vectors are the critical points (instances) of the training set on the planes H1 and H2; all vectors which are lying on one side of the hyperplane are labelled as 1 and the ones on the other side are labelled as -1. If $\mathcal{X}$ is a vector in a vector space and w is a weight coefficient vector and b is a bias term, the hyperplane H can be defined as:

$$\mathrm{x}_i \cdot \mathrm{w} + \mathrm{b} \geq +1 \text{ when } y_i = +1$$

and

$$\mathrm{x}_i \cdot \mathrm{w} + \mathrm{b} \leq -1 \text{ when } y_i = -1$$

in which $\mathrm{x_i} \cdot \mathrm{w} + \mathrm{b} = 0$ is the optimal hyperplane (H)

H1 and H2 are the planes where: H1: $\mathrm{X}_i \cdot \mathrm{w} + \mathrm{b} = +1$ and H2: $\mathrm{X}_i \cdot \mathrm{w} + \mathrm{b} = -1$. Note that the hyperplane A (grey line) is not an optimal hyperplane as, though it separates the classes, it only separates with the minor margins. The SVMs locate the hyperplane with the maximum Euclidean distance from the training set.

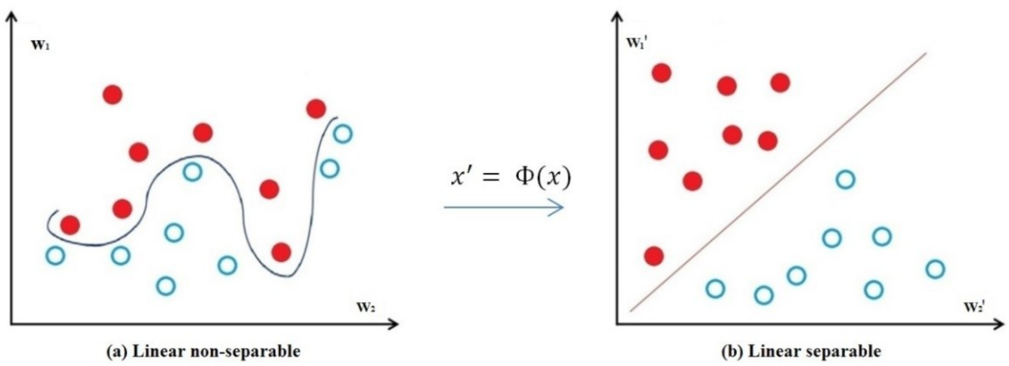


Figure 2.19: Feature mapping

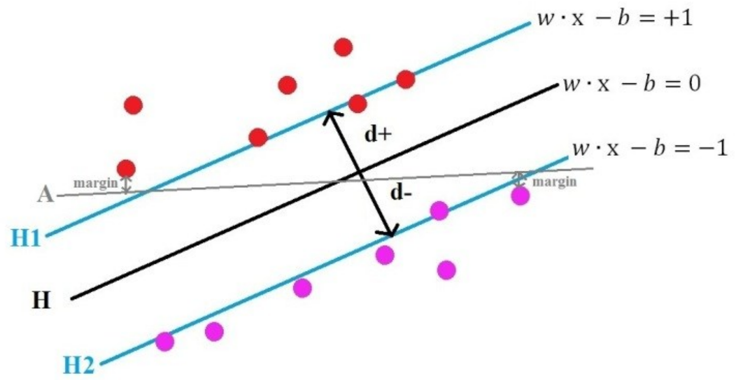


Figure 2.20: Separate hyperplanes in a two-dimensional space

When the SVMs are dealing with non-linearly separable data, the method called soft margin is used. As can be seen in Figure 2.21, the i^{th} misclassified sample is assigned a slack vaiables i (ξ_i); the ξ_i are positive variables which indicate tolerances of misclassification. So to minimise: $\mathrm{w} \cdot \mathrm{w} + C\,\Sigma_i\, \boldsymbol{\xi}_i^{\delta}$ where $\delta \geq 0$ under constraints: $y_i[\mathrm{w} \cdot x_i + b] \geq 1 - \boldsymbol{\xi}_i$ where $\boldsymbol{\xi}_i \geq 0$ and that $\boldsymbol{\xi}_i$ allow some errors, and C is the regularisation constant determining the trade-off between the missclassification rate for known training vectors (empirical risk) and the complex term; a higher C value corresponds to higher penalty errors. If C is too small, insufficient stress will be placed on fitting the training data. On the other hand, if C is too large, the algorithm will overfit the training data [173]. To minimise the number of errors, δ should be close to zero. A kernel is a function K, such that for all x,y $\in \mathcal{X}$, basic kernels which are used with the SVMs include:

Linear kernel: $K(x_i, x_j) = x_i^T x_j$

Polynomial kernel: $K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0$

Radial Basis Function kernel (RBF): $K(x_i, x_j) = \exp(-\gamma \| x_i - x_j \|^2), \gamma > 0$

Sigmoid kernel: $K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r)$

where γ, r, and d are kernel parameters.

The SVMs are able to solve two-class problems as well as multi-class problems. For multi-class problems, two common implementations of SMVs, namely One-Against-All (OAA) and One-Against-One (OAO), are used. OAA performs C binary SVMs to solve a problem with C classes; this involves constructing one SVM per class, which is trained to differentiate the samples of one class from the samples of the rest of the classes. The OAO builds one SVM for each pair of classes, or in the other word, it employs $C(C\text{-}1)/2$ SVMs in which each of the SMVs is trained to distinguish the samples of one class from the samples of another class. It was reported that OAO is more practical as the training process is quicker and yielded better results when compared to OAA. However, not every researcher agrees as some have advised that OAA strategy is as accurate as any other approach [174], [175].

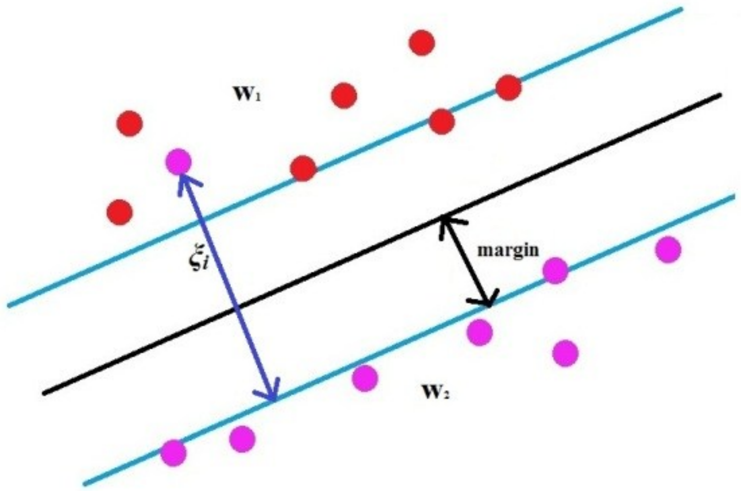


Figure 2.21: A non-linear separable training set

The SVMs have been successfully applied in pattern recognition applications such as face, speaker, and handwritting recognition [172]. For handwritting recognition, the SVMs have been employed in both off- and on-line handwriting recognition in various applications including segmentation, character, word, and signature recognition, and signature verification [173], [158], [176], [177], [178]. Papavassiliou *et al.* [177] employed SVMs in text lines and words segmentation work. Their word segmentation is based on a gap metric that exploits the objective function of a soft-margin linear SVM that separates successive connected components; the linear kernal was chosen rather than more complex kernels such as RBF because it would result in larger margins for connected components with significant vertical overlapping that most probably belong to the same word. The SVM was employed together with k-nearest neighbours (kNN) algorithm to create a kNN-SVM hybrid model used in handwriting cursive character recognition. Introducing the SVMs to their system improved the performance of kNN signficantly (from 1.00 – 3.61%) [158]. Another hybrid classifier system was proposed by Al-Boeridi *et al.* [178], they combined ANN and SVM classifiers together; the hybrid system was used to recognise Malay cheque words. In their system, the lexical word classification system enforces lexical matching on top of character recognition. The best recognition accuracy obtained at word level was up to 98.7%.

The SVMs have been applied in holistic based off-line handwriting as well as segmentation based. In the study proposed by Sagheer *et al.* [41], SVM with RBF kernel was applied in off-line Urdu word recognition and the best recognition rate of a 97% was obtained. The SVM was also employed in Kannada handwritten numeral recognition and attained 97.75% recognition rate [179]. Signature verification systems are also successfully benefited from employing the SVMs as the classifier; many researchers have employed the SVMs and obtained high accuracy rates with low average value of error rates in their work, [42], [43], [86], [180].

2.7.4 Hybrid Classifier

Hybrid classifier or multiple classifier is a method that combines different classifiers to solve the same problem; however, different errors are produced by each classifier. An appropriate combination of classifiers, therefore, should be able to yield a better reliable recognition rate than when only one classifier is applied. Hybrid classifiers have been employed in many studies. Common hybrid classifiers include, but are not limited to, HMM and ANN, ANN and SVM, KNN and SVM, and HMM and fuzzy logic [117], [155], [168], [175], [178]. In [117], the researchers employed HMM and fuzzy logic to recognise online Urdu script-based language. The recognition rate of 87.6% was obtained and was encouraging as it demonstrated increased the efficiency but at the same time the computational complexity. Zanchettin et al. [155] employed a hybrid classifier consisting of KNN and SVM to recognise cursive handwriting. The hybrid classifier yielded a substantial improvement in terms of recognition rate compared with MLP, KNN by itself (1.00 - 3.61%). An example of a hybrid HMM and ANN can be found in [168], where the hybrid classifier was used in off-line handwritten text line recognition. Hybrid HMM/ANN models compute the emission probabilities for the HMMs with a neural network instead of the commonly used Gaussian mixtures. The researchers reported that an impressive 42% relative improvement in word error rate over their baseline was obtained.

The employment of ANN and SVM hybrid classifier was proposed by Al-Boeridi et al. [175]. An off-line Malay character recognition system employing two individual classifiers, namely an adaptive multilayer feed-forward back-propagation NN and SVM. The study suggested that the result was promising and it could be further generalised to the entire Malay lexical dictionary in future work toward scaled-up applications.

Chapter 3

RESEARCH METHODOLOGY AND PROPOSED TECHNIQUES

The literature reviewed in Chapter 2 evidences that off-line handwriting recognition is still an active research in pattern recognition areas. This research thesis focuses on handwritten word recognition problems using a holistic approach. Even though the research in this area has been active for decades as reviewed in the previous chapter, it is still on-going, relevant, and challenging. Many applications apply off-line handwriting recognition techniques; these include bank cheque processing, postal address recognition, word spotting, form processing, signature recognition and verification, and automated assessment systems. New developments in feature extraction techniques and classifier adaptations are continuously taking place in order to improve recognition and accuracy rates, and the time used in recognition processes.

Despite the fact that computer-based assessment systems are now in use worldwide, paper-based assessment systems are still used in many parts of the world. This is due to the fact that in some countries, equipment is not always available and the cost could be considered expensive. Automatic assessment is one of the off-line handwriting recognition applications. Assessment of handwritten examinations is a difficult task; it requires the marker's concentration, precision and it is time-consuming. Having an automatic handwritten examination assessment system would obviously be advantageous for markers to overcome some of the aforementioned problems.

This chapter describes the methodology techniques employed in this research into an off-line short answer assessment with student identification system. This chapter is divided into 2 main sections which are: 3.1) the proposed off-line short answer assessment with student identification system and 3.2) a brief description of Thai language. For Section 3.1, each of the proposed sub-systems namely, Short answer Assessment System (SAAS), and Student Identification System (SIS) are described. Sub-section 3.1.1 details the OFAAS and sub-section 3.1.2 describes the SIS, both of the sub-system's proposed methodologies and techniques are described in detail under their section. Information with regards to Thai language can be found in Section 3.2.

3.1 Proposed System, Proposed Research Methodology and Proposed Techniques

The proposed Off-line Short answer Assessment with Student Identification System (OFSASIS), which in this case is a complete automatic short answer assessment system with student identification, consists of two sub-systems, namely a short answer assessment system and student identification system. Whereas the short answer assessment sub-system automatically marks off-line short handwritten examination papers, the student identification system identifies students from their handwritten first, middle (if applicable), and last name. Once both processes are completed, the full report containing a list of students who attended the examination together with the marks they obtained is produced. Each of the sub-systems, namely, the off-line short answer automatic assessment system and student identification system is described in Sub-section 3.1.1 and 3.1.2 respectively. Proposed methodology and techniques employed in each system are also described under each of the sub-section.

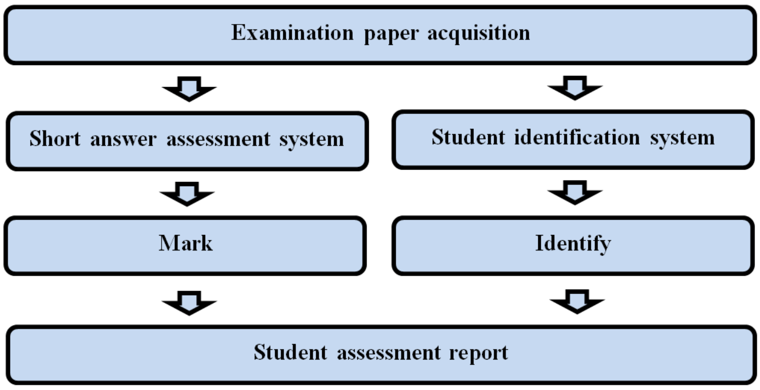


Figure 3.1: A block diagram illustrating a complete off-line automatic assessment system with a student identification system

3.1.1 Off-line Short Answer Automatic Assessment System

The proposed off-line Short Answer Assessment System (SAAS) which is one of the sub-systems of the proposed OFSASIS has the function of marking the short answer question examination paper. The proposed system can be considered challenging as it is one of the off-line handwriting recognition systems which have many disadvantages compared to on-line handwriting recognition systems. Whereas the on-line technique captures the real-time information of the writing, which is crucial in the recognition process, off-line techniques perform recognition based only on the scanned image, which does not have the same information as on-line techniques. Also, it can be even more difficult to recognise the handwriting of students when they have answered questions in examinations as they may be writing with stress and could be writing so fast that it reduces legibility. Aside from improving recognition rates and accuracy, the aim of this research was also to improve the automatic assessment systems' usability. The proposed system also includes assessment criterion to augment its accuracy and usability. Partly correct answers will also be marked.

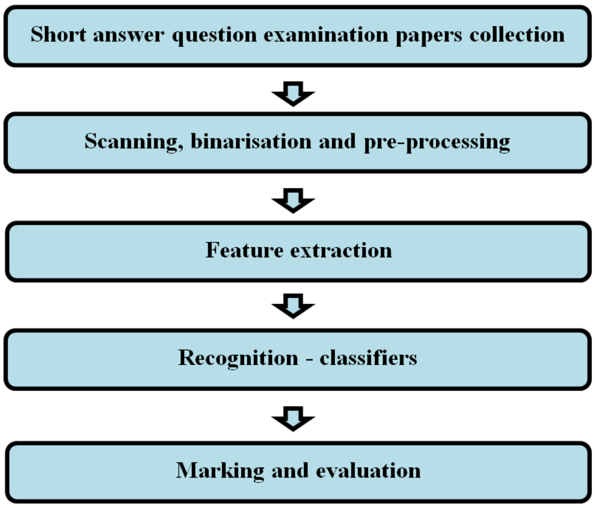


Figure 3.2: A block diagram illustrating the proposed off-line Short Answer Assessment System (SAAS)

As can be seen in Figure 3.1, the proposed SAAS which is one of the sub-systems has function of marking the short answer question examination paper. The obtained marks are then put into a report according to the student name obtained from the student identification sub-system. The proposed methodology includes data collection, image processing, pre-processing, and utilises the proposed feature extraction techniques in conjunction with different classifiers, which are Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs), in order to achieve the best results and for comparison purposes. The block diagram in Figure 3.2 illustrates the proposed system methodology and processes. The individual components of the system are discussed as follows.

A. Data Collection and Datasets

The proposed system is primarily concerned with assessing short answer questions from examination papers. The examination used to evaluate the proposed system is a short answer question assessment with no given answers to select (i.e. not multiple choices). The questions are concerned with fundamental information technology knowledge. Due to the questions being closed questions, there can only be one correct answer for each question. For example, the answer to the question "What does IT stand for?" can only be "Information Technology". Other examination questions and their answer examples are:

1. What is the "brain" of the computer called?

Answer: CPU, Central Processing Unit.

2. What is another way to write 1,000,000,000 bytes?

Answer: One gigabyte, 1 gigabyte, a gigabyte.

3. What does PC stand for?

Answer: Personal Computer.

There is no publicly available examination answer dataset; as a result, a data collection process was performed to create a custom dataset. The number of examination papers was set at 52, and the answers to the examination questions were written by 52 students. The number 52 was determined based on an assumption that for most classes, the number of students per class is less than 50, except for larger classes such as some university lectures. The examination papers included 52 Hand-printed (HP), and 52 cursive handwritten (CH) samples from both male and female participants. Altogether, 104 examination papers containing ten short answer questions were used for conducting the experiments. All samples were written with minimum constraints (e.g. writing instruments were not restricted). As the samples collected from the examination papers contained some incorrect answers to each question, additional data collection of each missing sample also took place. For example, the correct answer of question number one is "information"; however, not all of the 52 examination papers contained the correct answer. Therefore, the word "information" needed to be further collected so that sufficient samples of each word could be available for training. The number of additional words collected for each question varied from 2 to 26 words.

For the datasets used in the proposed system, both HP and CH word samples used to facilitate ANN and SVM training and testing, about 80% of the available samples of each question were used as the training dataset, and about 20% were used as the testing dataset. The datasets used in the experiments include HP, CH, and hand-printed and cursive handwritten words combined (HPCHC). In the experiments, the handwriting datasets were each divided into two sub-datasets. Type I contained correctly spelt correct answers in both training and testing datasets. Type II contained some misspelt or wrong answers as well as correctly spelt correct answers in the testing dataset, whereas the training dataset only contained correctly spelt correct answers. Dataset Type I was used for feature extraction techniques performance comparison, and the Type II dataset was used for SAAS evaluation. The dataset Type I was only used for performance comparison to the Dataset Type II as it only contained correct answers to the questions, and therefore, was not suitable to be used for SAAS evaluation. For HP and CH datasets, 492 words were used for training and 132 words were used for testing. For HPCHC, the training dataset contained 984 words and the testing set contained 264 words.

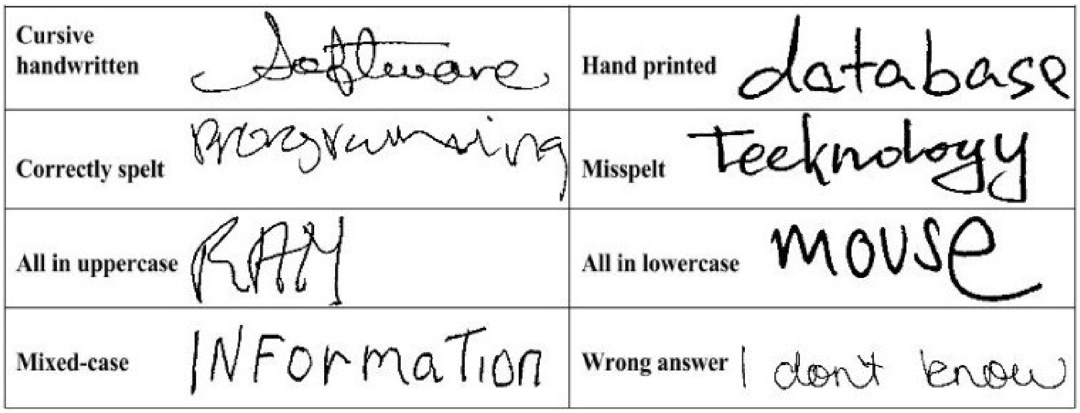


Figure 3.3: Categorised short answer examples

It is to be noted that for the training dataset, all of the samples of each question were the correct answers to the questions, and were correctly spelt. However, not all samples which were collected from the actual examination papers were the correct answers to the questions, as well as not being correctly spelt. Therefore, some of the samples in the training sets were obtained from the examination papers, and some additional samples of each word were obtained later to add to the training set so that the correctly spelt 984 words (about 80% of the full dataset of 1,248 words) could be used for training.

The samples were collected with as few constraints as possible, apart from students being asked to write one paper in HP style, and the other in CH. Samples which were used in this research were varied. The dataset can be divided into five variant types which are: correctly spelt, incorrectly spelt, all in uppercase, all in lowercase and mixed-case. For instance, the answer to the question “What does IT stand for?” is “Information Technology”. However, the answers which were found were “INFORMATION TECHNOLOGY”, “information technology”, “Information Technology”, “Information technology”, “info technology”, misspelt “tecknology", and “INFormaTion TecHnoLoGy”. Categorised short answer examples can be found in Figure 3.3.

B. Image Acquisition and Preprocessing

All examination papers were scanned with 300 dpi resolution and stored in grey-level format, and then were binarised. All examination papers were scanned as 256 grey-scales by using a flatbed scanner with 300 dpi solution. The images were later converted in to a .pbm (Portable bitmap) file format. In this research the images were binarised in the preprocessing phase. Whereas the white is the background, the black is the foreground. Hence the words are in black and the paper is in white. Binarising the images is to convert the characters or black pixel into bit 1’s, and the white background or paper into bit 0’s. By doing so, the other preprocessing techniques could be performed more easily [181]. Line and word segmentations were performed. A slant normalisation process was not included; however, the images were skew normalised. Most of the images still had noise and therefore some of the strokes were still broken. The images were used as they were. For each image, boundary extraction was performed to isolate connected components. Subsequently, loop as well as upper and lower contour extraction was performed. Upper and lower contour extraction was performed in order to find upper and lower reservoirs only and not to be used as features themselves; therefore, a skeletonisation approach was not considered in this research. An example to illustrate the successive stages of word image preprocessing is presented in Figure 3.4; preprocessing techniques, which are binarisation, noise removal, line and word segmentation, skew normalisation, boundary extraction, upper and lower contour extraction, and loop extraction, are explained in details as follows.

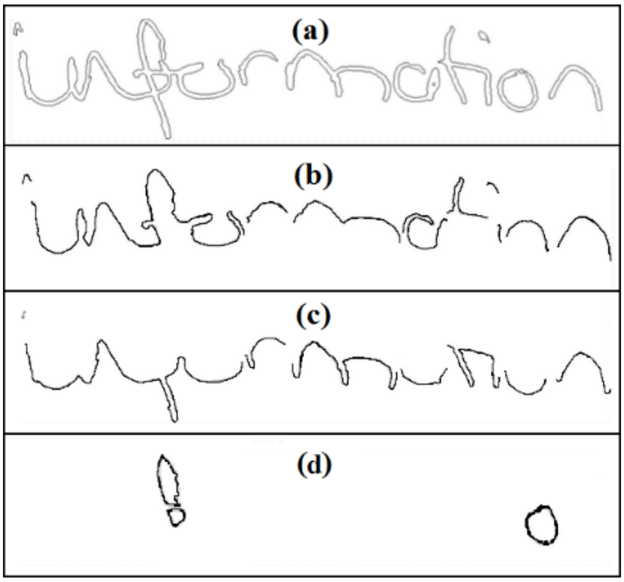


Figure 3.4: Example of images obtained after each preprocessing step (a) full boundary image (b) upper contour, (c) lower contour, (d) loops

After scanning each image was binarised using Otsu's algorithm [29], which only utilises the zeroth- and the first order cumulative moments of the grey-level histogram. After the document was scanned and binarised, salt and pepper noise was introduced. Based on this reason, the noise removal procedure as described in Algorithm 3.1 was performed up to four times on each image then the filling process (Algorithm 3.2) was performed. The procedures were performed to ensure that the images were clean and ready for the segmentation process. As a result, the images were considered clean and therefore line and word segmentation processes could be performed efficiently.

The next procedures were line and word segmentation. The segmentation algorithm used in this study is simple and straightforward, yet was still useful in this study. The answers to the examination used in this research were generally short with a few words per question. Since most of the images did not contain touching lines or words, the segmentation algorithms (Algorithm 3.3 and 3.4) implemented only dealt with white spaces between lines and words. As can be seen from examples of examination papers in Figure 1.2 on page 4, each line was well separated from the other lines and each word was well separated from the other words. Also since noise removal and filling techniques were performed prior to the line segmentation process, the segmentation algorithm employed was able to segment each line efficiently. As a result, words with ascenders and descenders did not affect the ability of line segmentation, and therefore did not affect the experiment results.

After the segmentation process was performed on all images, each image was manually checked to see if it was segmented correctly. If the images were not segmented properly, manual segmentation was performed on these images; this was to ensure that all the images were completed (no missing parts of letters or missing letters), and that the best recognition rates could be obtained. Noise removal, filling, line and word segmentation algorithms can be found in Algorithm 3.1, 3.2, 3.3 and 3.4 respectively.

After word segmentation was performed, skew normalisation took place. Skew normalisation was performed so that each image was in the best position possible on the base line (see Figure 3.5). Skew normalisation technique [94] was employed in this research; the algorithm (Algorithm 3.5) is described as follows:

Algorithm 3.1 Noise Removal

```
while there are Columns and Rows do
  Inspect Column Pixel whether It is Black or White
  if the Current Pixel is White
     Move to the Next Column Pixel
  else if the Current Pixel is Black
     Count the Number of Black Pixels around the Current Pixel
     if the Number of Black Neighbour Pixels is Equal or Less Than Two
        Remove the Current Pixel
     end if
  end if
  end if
end while
```

Algorithm 3.2 Filling

while there are Columns and Rows **do**
  Inspect Column Pixel whether It is Black or White
  if the Current Pixel is White
    Count the Number of Black Pixels around the Current Pixel
    if the Current Pixel is Surrounded by Eight Neighbours Pixels
      if the Number of Black Neighbour Pixels is Equal or More Than Five Pixels
        Fill the Current Pixel
      end if
    end if
    else if the Current Pixel is Surrounded by Less than Eight Neighbour Pixels
      if the Number of Black Neighbour Pixels is Equal or More Than Three Pixels
        Fill the Current Pixel
      end if
    end if
  end if
end while

Algorithm 3.3 Line Segmentation

while there are Rows **do**
  Inspect Row's Pixel Density
  if Pixel Density is Zero
    Segmentation Point Found
  end if
end while

Algorithm 3.4 Word Segmentation

```
while there are Columns do
  Inspect Column's Pixel Density
  if Pixel Density is Zero
      Count Number of Columns that the Pixel Density is Zero
end while
Calculate the Average Number of Zero Density Columns
while there are Columns do
  Inspect Column's Pixel Density
  if Pixel Density is Zero
      Count Number of Columns that the Pixel Density is Zero
      if the Number of Columns that the Pixel Density is Zero is greater
      than Average
          Segmentation Point Found
      end if
  end if
end while
```

Algorithm 3.5 Skew Normalisation

```
Ascender and descender removal
Word image centre is located and is cut vertically into two equal windows
The centre of gravity (centroid) for each window is located
Hypothesise a line that joins the two sets of coordinates (centroids of the two windows)
Calculate the slope (angle of skew) of the word
Employ a geometric operation for the rotation
```

The next preprocessing step was to extract the word boundary as connected component analysis was used in feature extraction process. Upper, lower, and loop extraction was also performed on word boundary extracted. The boundary extraction was done by employing the algorithm described in [61]. The extraction was started by the image foreground pixels which were searched by using raster scan. Once a foreground pixel is found, its foreground neighbour pixels are counted. If the number of the neighbour pixels is less than three, it is marked as boundary pixels. All the pixels which have not been marked are converted into background pixels. For the remaining pixels, the number of adjacent boundary pixels is counted. If the pixel has more than two foreground neighbours, it will be altered into background pixels. This process is repeated until no further change occurs; once the process is finished, the boundary extracted image is obtained. Boundary extracted image was further used to find important features such as WR, loops, location transitions and direction transitions, which may help to increase recognition and accuracy rates. The boundary extraction algorithm can be seen in Algorithm 3.6 and a word boundary extracted sample can be seen in Figure 3.6.

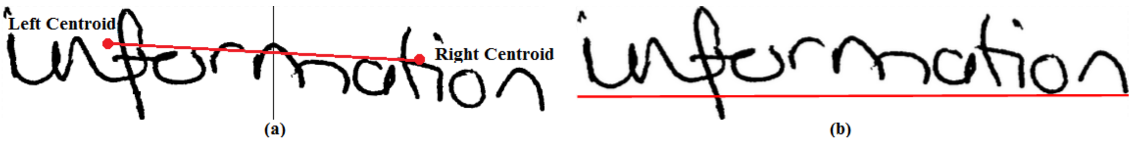


Figure 3.5: (a) Centroid locations and a hypothesised line and (b) a skew normalised image

After the boundary extracted image of each word image was obtained, loop, upper and lower contour were extracted. Loops are an important aspect of a word; they help to recognise characters or words which contain loops in the image. In 'b' and 'l', for example, the loop feature will help classify 'b' from 'l'. To be able to extract upper and lower contours, which were used later in the feature extraction phase, the loops needed to be located first. The reason for this being that sometimes the word/character stroke is so thin that the loop and upper or lower contour shares the same pixel; this may cause a problem when trying to locate water reservoirs from each of the contours. From Figure 3.7, it can be seen that to locate a loop, contour information, namely outer boundary contour, inner boundary contour, top-, bottom-, left-, and right-most pixels, is used to locate the loop. In this research, only complete loops are used as features; therefore incomplete loops such as "A" in Figure 3.7 are ignored. Algorithm 3.7 explains how loops are located in a word image.

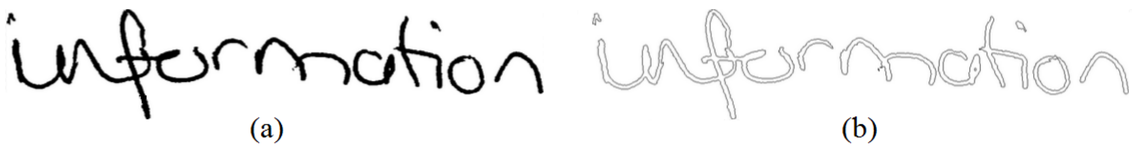


Figure 3.6: (a) Binarised image and (b) its boundary extracted image

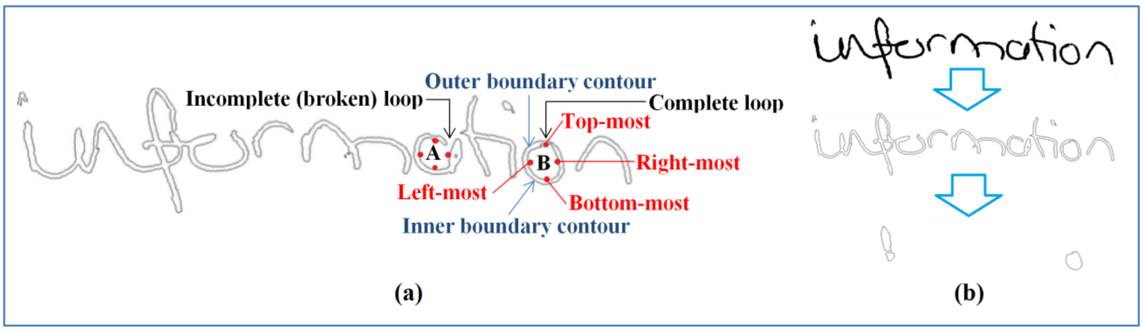


Figure 3.7: (a) Loop extraction - A is an incomplete loop and B is a complete loop with outer and inner boundary contour information; (b) a binarised image and its boundary extracted image and the loops extracted

Algorithm 3.6 Boundary Extraction

for each foreground pixel *p* **do**
    if *p* has less than three foreground pixel neighbours
        mark *p* as boundary pixel
    else
        mark *p* as background pixel
    end if
end for
done ← false
while not done **do**
    done ← true
    for each boundary pixel *p* **do**
        if {boundary pixel *q* adjacent to *p*} 2 **then**
            mark *p* as background
            mark boundary neighbours of *p* as boundary
            done ← false
        end if
    end for
end while

Algorithm 3.7 Loop Extraction

```
for each column do
  for each row do
    scan for a foreground (black) pixel
      if found a foreground pixel (assuming to be an outer contour)
        search for the next foreground pixel of the same column
          if found a foreground pixel (assuming to be an inner contour)
            go to the next row and check if the next row of the same column is a
            background pixel (to check if it is actually an inner contour)
            if an inner boundary contour is found
              if the inner boundary pixel coordinates are the same with the ones which
              has already been recorded as the loop then
                skip
              else
                search for top-most, bottom-most, left-most, and right most pixel
                if the four pixels are found then
                  connect the four points using 8-directional chain code
                  if the four pixels can be connected then
                    record the loop location and its coordinates
                    write the loop into an image of loops of the word image
                  end if
                end if
              end if
            end if
          end if
      end if
  end for
end for
```

Once the loops of each image are found and written as a separate file, the next steps are to find the image lower and upper contour. To find the upper contour, first, scanning is performed from the first row of each column until the first foreground pixel is found. Once the pixel is found its coordinates are recorded, and then the scanning moves to the next column. The scan is continued until it has finished the last column (image width); from the first pixel coordinates of the left-most column, then attempt to connect it with the consecutive (column) pixel coordinates if applicable by employing an 8-directional chain code. To obtain the lower contour, the similar procedures are applied; the difference being the search is started from the last row up to the first row. Images of lower and upper contour are written in separate files for further processes.

Examples of upper and lower contours extracted from the full boundary image can be seen in Figure 3.8.

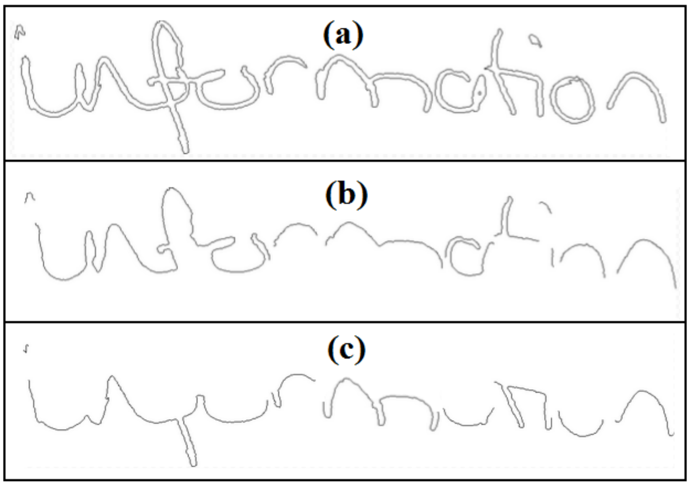


Figure 3.8: (a) Full boundary contour, (b) upper contour extracted, and (c) lower contour extracted images

C. Feature Extraction

Feature extraction is one of the most crucial aspects of a handwriting recognition system. The objective of feature extraction is to extract the salient information that needs to be applied in the recognition process. It reduces data dimensionality by determining certain feature properties that distinguish input patterns [16]. Feature extraction techniques employed in this research are: 1) the modified direction feature, 2) Gaussian grid feature, 3) water reservoir feature, 4) loop feature, 5) centre of mass, and proposed water reservoir, loop and Gaussian grid feature extraction technique. Feature extraction techniques employed in the proposed off-line Short Answer Assessment System (SAAS) and their corresponding sizes can be seen in Table 3.1.

1) The Modified Direction Feature (MDF) Extraction Technique: the MDF has been employed in a number of studies, and has been reported to yield positive results for both signature verification and character recognition work [182], [82], [183], [184], [185]. For this reason, this study employed a technique to examine whether the MDF is efficient to be used in whole word recognition rather than character recognition and signature verification; also it aimed to explore the way to improve the performance of the original MDF itself.

Originally, the MDF was created to extract direction and location information at background to foreground transitions from handwritten characters. Hence, the technique was developed to analyse information at the character level. However, the proposed system will be implementing the MDF to extract information from the whole word image. Likewise, the recognition process will utilise holistic name information rather than recognising the handwriting at the character level. The MDF technique builds upon direction feature extraction.

The main difference in MDF is the way the feature vector is created. More details regarding direction feature and the MDF are presented below.

- **Direction Feature (DF)** extraction technique was proposed by Blumenstein *et al.* [73]; it sought to simplify each character's boundary or thinned representation through identification of individual stroke or line segments in the image. After that the new character representation was broken down into a number of zones of equal size whereby the number, length and types of lines present in each window was determined. The line segments which would be determined in each character image were categorised into four types namely, vertical lines, horizontal lines, right diagonal and left diagonal. The DF also located intersection points between each type of line. The character pattern was prepared by the following steps:

1. Starting point and intersection point location: because for cursive English handwriting, many letters begin in the lower, left hand side, to locate the starting point of the character therefore, the first black pixel in the lower left hand side of the image is selected to be the starting pixel. After that the intersection points between line segments, which are determined as being those foreground pixels that have more than two foreground pixel neighbours, are marked.
2. Distinguish individual line segments: neighbouring pixels along the thinned pattern boundary are followed from the starting point to known intersection points. At each subsequent intersection, the algorithm conducts a search in a clockwise direction to determine the beginning and end of individual line segments. As the result, the commencement of a new line segment is located if one of the following conditions is true: 1) the previous direction was up-right or down-left AND the next direction is down-right or up-left 2) the previous direction is down-right or up-left AND the next direction is up-right or down-left 3) a line segment direction has been changed more than three types 4) the length of the previous direction type is more than three pixels.
3. Labelling line segment information: the black pixels along the length of this segment are coded with a direction number being '2' for vertical line segment, '3' for right diagonal, '4' for horizontal line segment, and '5' for left diagonal line segment.
4. Line type normalisation: the number of values belonging to each direction type are tallied in a particular line. The spurious direction values are replaced by the direction value most frequently represented in a particular line segment.

- **The Modified Direction Feature (MDF)**

For MDF, feature vector creation is based on the calculation of transition features from background to foreground pixels in the vertical and horizontal directions. Both the location transitions (LTs) calculated, and the direction value at that location are stored. The feature extraction processes includes 1) Determining LT values, and 2) Determining Direction Transitions (DTs).

To determine LT values, each row of the image is scanned from left to right and right to left. Each column in the image also must be scanned from top to bottom and vice versa. A fraction value of the distance traversed across the image is attained by computing the LT values in each direction. A maximum value is defined to be the largest number of transitions that may be recorded in each direction. However, if there are less than the maximum transitions recorded, the remaining maximum value – *n* transitions will be assigned values of zero.

DT values can be found after a transition in a particular direction is found. The direction value at the position is stored along with its corresponding LT value. DT values are obtained by dividing the direction value by a predetermined number. As a result, four vectors would be present for each set of feature values which makes a total number of eight vectors altogether. Another process which has to be performed is re-sampling. Re-sampling of the vector is essential to ensure that the dimensions are normalised in size. An example of processing of DT and LT values in the left-to-right direction can be found in Figure 3.9. Further information regarding MDF can be found in [185].

As mentioned earlier, in the proposed system, the MDF will be implemented to extract information from the whole word (each name component) image. Likewise, the recognition process will utilise holistic name information rather than recognising the handwriting at the character level. The vector sizes are 160 or 120 when applying MDF to the full boundary image, the vector size is determined by employing the formula:

noOfRowFeatures × noOfTransitions × noOfVectors × resampledMatrixHeight(Width)

Where:

noOfRowFeatures = 2,

noOfTransitions = 3 or 4,

noOfVectors = 4, and

resampledMatrixHeight(Width) = 5

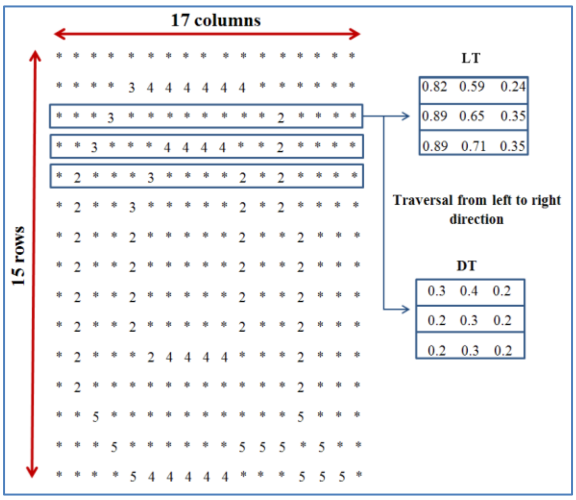


Figure 3.9: Processing of DT and LT values in the left-to-right direction

In [82], the MDF element feature vector size was increased from 120 to 121, where the additional feature is the ratio is a global feature that considers the proportion of the height and the width of the image. The system proposed employed the MDF with 121 element feature vector. The MDF was employed on full boundary images. Figure 3.9 illustrated the processing of DT and LT values in the left-to-right direction

2) The Gaussian Grid Feature (GGF) Extraction Technique: the Gaussian Grid Feature [86] is a relatively new feature extraction technique and is a grid-based feature extraction technique. Originally, it was developed for the signature verification problem and the results with an average error rate of 13.90% with only 0.02% false acceptance rate for random forgeries from employing the technique are encouraging. The GGF employs signature (pattern) contours as its input. From the contour representation of a name component image, the GGF extraction technique performs as follows:

1. The contour image is divided into $m \times n$ zones of equal size.
2. The 4-direction chain code histogram of each block is created by tracing the contours in each block.
3. A Gaussian smoothing filter (σ=1.2) is then applied to each directional $m \times n$ matrix.
4. The value of each element of each matrix obtained in the previous step is then normalised by dividing its value by the maximum value of the four matrices.
5. From the two-matrix pairs, horizontal (H) and vertical (V) matrices, left-diagonal (L) and right-diagonal (R) matrices, two new matrices ($\oplus$ and $\otimes$) are established by manipulating pairs of matrices of the perpendicular directions.
6. The feature vector is formed by merging the six matrices H, V, L, R, $\oplus$, and $\otimes$. The dimension of the output feature vector is $12\times12\times6=864$.

In the research proposed here the grid configuration size is 12×12 is employed on each word image; therefore, the vector size is 864.

3) Water Reservoir Feature (WRF) Extraction Technique: generally, the Water Reservoir (WR) feature was used in segmentation tasks in both numeric and word segmentation [123], [125]. However, in the research proposed here, the technique was used to locate WRs found in upper and lower contours of images (refer to Figure 3.10). After the reservoirs are found, the features of these reservoirs are extracted. WR features were used in this research because the English language has this property in its script. WR properties can be seen in Figure 3.11.

The WR principle supposes that if water is poured from the side of a script component, the cavity regions of the background portion of the component where water will be stored are considered as reservoirs of the component. To find water reservoirs, first upper and lower contours are found. The top reservoir (located at the upper contour) is found if the water is poured from the top of the component, and the bottom reservoir (located at the lower contour) is found where the water is poured if rotating the component by 180°.

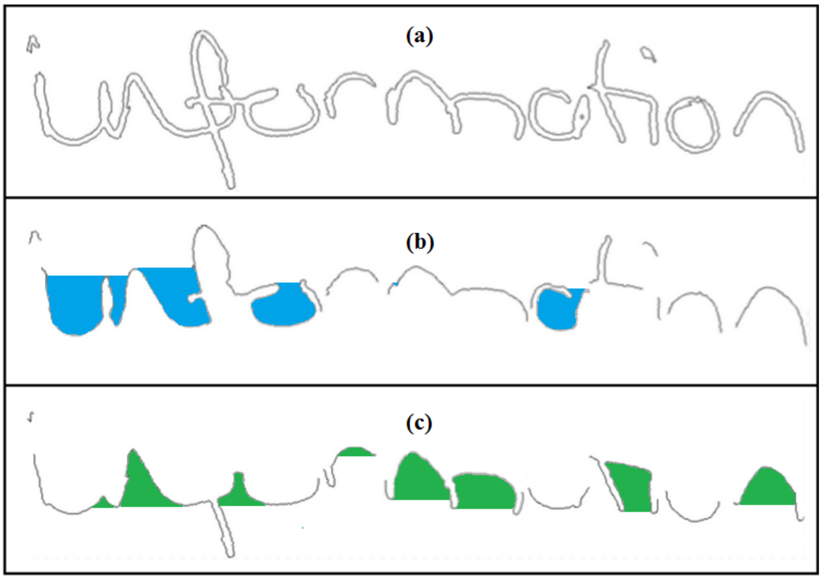


Figure 3.10: (a) The full contour image (b) top WRs retrieved from the upper contour, and (c) bottom WRs retrieved from the lower contour

For each contour image, the image was divided into seven equally-sized windows. In each window, once each WR was located (if any), seven WR features were extracted. The seven WR features used in this research are:

- WR area which is the area of the cavity region where water will be stored (1 feature).
- The number of pixels inside a reservoir is tallied and is considered the area (1 feature).
- WR level that overflows from a reservoir (1 feature).
- WR flow side is where the water flows from; this can be left, right or either side, if the flow level is the same for both sides (2 features).
- WR base line (Y co-ordinate) (1 feature).
- WR width and height (2 features).

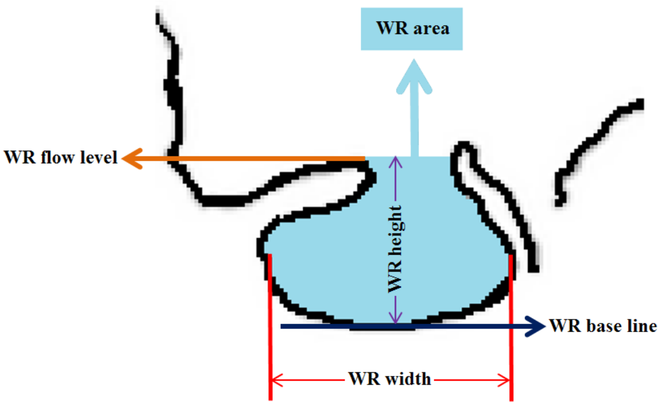


Figure 3.11: Water reservoir properties

The ratio between width and height was applied to each feature as none of the images were size normalised. The number of WRs is set to four per window based on the assumption that there are less than four WRs per window. This makes the WR feature vector 196 (7 windows × 4 WRs × 7 features) in size per contour image, and so makes the vector size equal to 392 for both upper and lower contour images.

4) Loop Feature (LF) Extraction Technique: as the nature of English language may contain loops in each character and because for English language, the loops play an important role in distinguishing characters for example ‘l’ and ‘b’, the research proposed here also extracts features from loops found in images (see Figure 3.12). In order to obtain loop features, loop images are first divided into 3 zones (baseline, middle, and top zone). Then loops are located within each zone. Each zone allows fifteen loops to be stored in a feature vector. Each loop has four features being the loop area, which is obtained from the total number of pixels inside a loop, loop height, loop width and the zone that the loop is located in. There are two sizes of loop vector being 120 and 192 employed in some experiments; for both sizes in order to obtain loop features, loop images are first divided into 3 zones (baseline, middle, and top zone). Then loops are located within each zone. The explanations of each vector size are as follow:

i. **Loop vector of 120 elements:** each zone allows nine loops to be stored in a feature vector. Each loop has four features being the loop area, which is obtained from the total number of pixels inside a loop, loop height, loop width and the zone that the loop is located in. Therefore, for the dataset investigated, the loop feature vector size is 120; the total number of 120 comprises 3 zones × 9 loops from each zone × 4 features of each loop + 12 additional features for the average loop area, average loop width and height, and total number of loops of each zone.

ii. **Loop vector of 192 elements:** Each zone allows fifteen loops to be stored in a feature vector. Each loop has four features being the loop area, which is obtained from the total number of pixels inside a loop, loop height, loop width and the zone that the loop is located in. Therefore, for the dataset investigated, the loop feature vector size is 192; the total number of 192 comprises 3 zones × 15 loops from each zone × 4 features of each loop + 12 additional features for the average loop area, average loop width and height, and total number of loops of each zone.

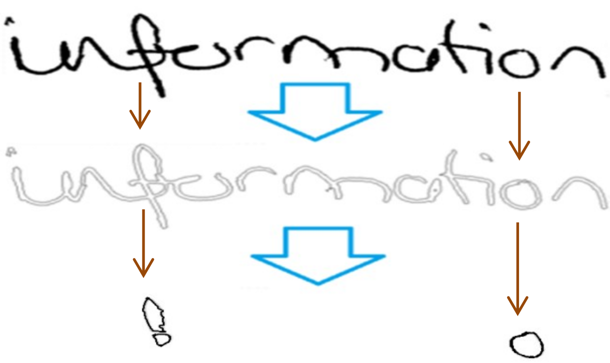


Figure 3.12: Loop extracted

5) Water Reservoir, Loop, and Gaussian Grid Feature (WRLGGF) Extraction Technique: the proposed WRLGGF was developed based on three feature extraction techniques being Water Reservoir Feature (WRF), Loop Feature (LF) and GGF. WRF and LF can be found in English language, and are important character identifiers for the English language, for example, 'd' from 'l'. WRF and LF were combined with GGF in order to extract the most salient information, so that the best recognition rates possible could be achieved. From full contour images, the process of top contour, bottom contour, and loop finding was undertaken. From full boundary images, there were three that were retrieved, being top contour image, bottom contour image, and loop images (see Figure 3.13). For the top and bottom contour, each reservoir was located, then reservoir feature extraction was applied on each reservoir and recorded in a feature vector. Once top and bottom reservoir extraction processes were completed, loop feature extraction was processed and features retrieved from each loop were recorded in the vector. Finally, GGF was then applied on the full boundary contour to extract features from the image. There are two final WRLGGF vector sizes being 1) size of 1,448 (392 from WRF + 192 from LF + 864 from GGF), and 2) size of 1,376 (392 from WRF + 120 from LF + 864 from GGF).

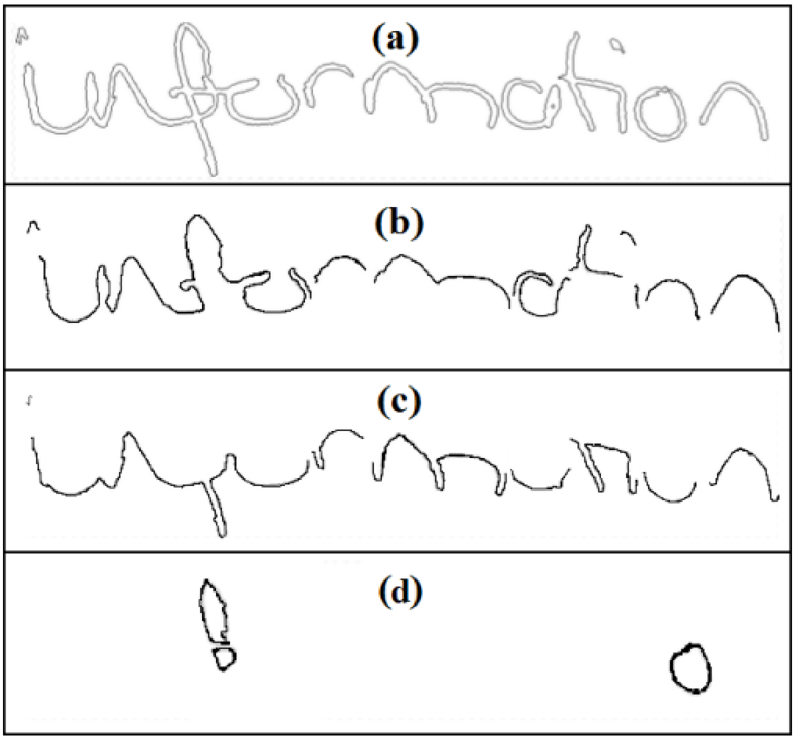


Figure 3.13: The three images being (b) upper contour, (c) lower contour, (d) loops extracted from (a) their full boundary image

6) Centre of Mass Feature (COMF) Extraction Technique: COMF is also known as centre of mass or centroid. The position of the COG can be calculated by [94]:

xCOG	= sum$(x_{ij} * i)/n$
i	= 0,.., NC-1 where NC is the width of the component
j	= 0,.., NR-1 where NR is the height of the component
x	= {0,1} Indicates the value of the current pixel being examined
n	= Number of foreground pixels
yCOG	= sum$(y_{ij} * j)/n$
i	= 0,.., NC-1 where NC is the width of the component
j	= 0,.., NR-1 where NR is the height of the component
y	= {0,1} Indicates the value of the current pixel being examined
n	= Number of foreground pixels

The centre of gravity therefore may be represented by: CoM = (xCOG, yCOG)

For the proposed system, with each image window, COG will be found and its coordinates (x, y) will be stored, and after will be used in the feature vector for the classifier. The COMF that was utilised here was extracted from four, eight, and nine images as can be seen in Figure 3.14. There were four sets of images used for COMF extraction. The 1st set (Set A) comprised two equal horizontal and two equal vertical windows which were obtained from each full image (see Figure 3.14 a). The 2nd set (Set B) comprised four equal windows which were obtained from dividing each full image into four equal windows (see Figure 3.14 b). The 3rd set (Set C) comprised eight images, four of which were from Set A and the other four were from Set B. The last set (Set D) comprised images from Set A and Set B combined with its full image (see Figure 3.14 f).

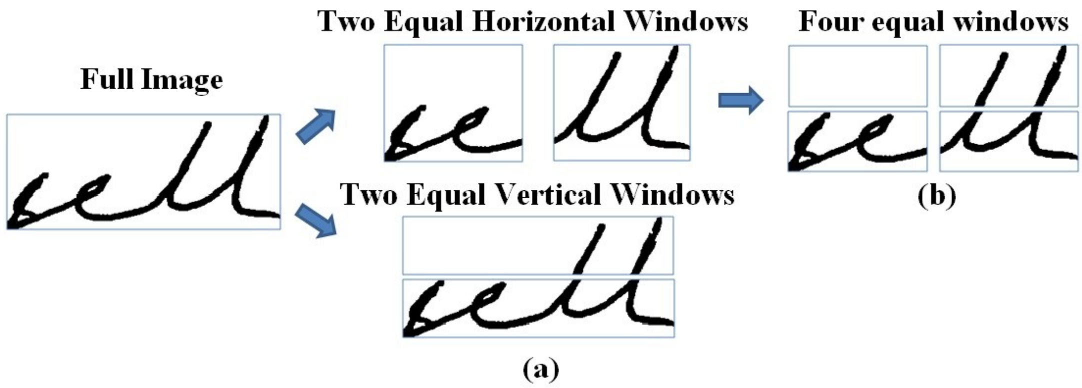


Figure 3.14: The eight windows obtained from a full image

For each window, X and Y centroids were calculated, and regarded as two features. Therefore, Set A and B vector size were 8 (VECT A and VECT B respectively); Set C had a vector size of 16 (VECT C); and Set D had the vector size of 18 (VECT D). Note that for Set D, the full image height and width ratio (P) was also applied on the full image centroids (CentroidX * P, CentroidY * P) as none of the images were resized.

7) The Enhanced Water Reservoir, Loop, and Gaussian Grid Feature (EWRLGGF) Extraction Technique: The original WRLGGF was based on three feature extraction techniques namely WRF, LF and GGF. WRF and LF were combined with GGF in order to extract the most salient information, so that the best possible recognition rates could be achieved. The EWRLGGF is divided into four new sub-extraction techniques, namely EWRLGGF_A, EWRLGGF_B, EWRLGGF_C, and EWRLGGF_D. The EWRLGGF_A and EWRLGGF_B were obtained by combining the WRLGGF with COMF VECT A and B respectively; this resulted in vector size of 1,456 (392 from WRF + 192 from LF + 864 from GGF + 8 from COMF VECT A and B respectively). For the EWRLGGF16_C, the feature vector was obtained by combining the WRLGGF with COMF VECT C which resulted in vector size of 1,464 (392 from WRF + 192 from LF + 864 from GGF + 16 from COMF VECT C). Lastly, EWRLGGF_D vector size is 1,466 (392 from WRF + 192 from LF + 864 from GGF + 18 from COMF VECT D)

8) The Enhanced Gaussian Grid Feature (EGGF) Extraction Technique: The EGGF is divided into four new sub-techniques, namely EGGF_A, EGGF_B, EGGF_C and EGGF_D. The EGGF_A and EEGGF_B were obtained by combining GGF with COMF VECT A and B respectively which resulted in vector size of 872 (864 from GGF + 8 from COMF VECT A and B respectively). For the EGGF_C, the feature vector was increasing to 880 (864 from GGF + 16 from COMF VECT C). Lastly, for the EGGF_D, the feature vector size was 882 (864 from GGF + 18 from COMF VECT D).

9) The Enhanced Modified Direction Feature (EMDF) Extraction Technique: The EMDF is divided into four new sub-techniques, namely EMDF_A, EMDF_B, EMDF_C and EMDF_D. The EMDF_A and EMDF_B were obtained by combining the MDF with COMF VECT A and B respectively which resulted in vector size of 129 (121 from GGF + 8 from COMF VECT A and B respectively). For the EMDF_C, the feature vector size was 137 (121 from MDF + 16 from COMF VECT C), and EMDF_D had the vector size of 139 (121 from MDF + 18 from COMF VECT D)

Table 3.1: Feature extraction techniques employed in the proposed off-line Short Answer Assessment System (SAAS) and their corresponding sizes

Feature Extraction Technique	Feature Vector Size
Modified Direction Feature (MDF)	121
Gaussian Grid Feature (GGF)	864
Water Reservoir Feature (WRF)	392
Nine Loop Feature (LF – 9)	120
Fifteen Loop Feature (LF – 15)	192
Water Reservoir, Loop, and GGF (WRLGGF)	1,448
Centre of Gravity Feature – Set A (Enhanced A)	8
Centre of Gravity Feature – Set B (Enhanced B)	8
Centre of Gravity Feature – Set C (Enhanced C)	16
Centre of Gravity Feature – Set D (Enhanced D)	18
Enhanced Water Reservoir, Loop, and GGF – Set A (EWRLGGF_A)	1,456
Enhanced Water Reservoir, Loop, and GGF – Set B (EWRLGGF_B)	1,456
Enhanced Water Reservoir, Loop, and GGF – Set C (EWRLGGF_C)	1,464
Enhanced Water Reservoir, Loop, and GGF – Set D (EWRLGGF_D)	1,466
Enhanced Gaussian Grid Feature – Set A (EGGF_A)	872
Enhanced Gaussian Grid Feature – Set B (EGGF_B)	872
Enhanced Gaussian Grid Feature – Set C (EGGF_C)	880
Enhanced Gaussian Grid Feature – Set D (EGGF_D)	882
Enhanced Modified Direction Feature – Set A (EMDF_A)	129
Enhanced Modified Direction Feature – Set B (EMDF_B)	129
Enhanced Modified Direction Feature – Set C (EMDF_C)	137
Enhanced Modified Direction Feature – Set D (EMDF_D)	139

It is important to note that each feature extraction technique employed different input images dependent on what kind of input image it employed. Specific features such as Water Reservoir (WR) feature was extracted from upper and lower contour images while loop features were extracted from loop images. Such input images were extracted in the manner which prevented the correlated features being extracted by each feature extraction technique (i.e. the loop images only contained complete loop images – closed loop, and therefore each loop was an actual loop not just only a part of contour). See the Figure 3.13 for image (b) upper contour image and (c) lower contour image; image (b) and (c) were employed by WRF extraction technique, while image (d) loop image was employed by the LF extraction technique.

It can be noted that upper contour (b), lower contour (c), and loop images (d) were extracted from their full boundary contour image (a). It was also possible that loops and upper and lower contours shared some similar pixels; this could occur in the case where the written stroke was thin (i.e. one pixel wide or high). However, as described earlier, since loops were considered as loops when they were closed loops, there was no problem with the shared pixels when extracting the images from their full boundary contours. And therefore, there was no problem with features being correlated.

D. Classification

From the 1990s, learning-from-examples based pattern classification methods have been broadly applied to character recognition; significant improvements of recognition accuracies have been made since. Support vector machines, statistical methods, artificial neural networks, and hybrid classifiers are examples of classification methods [186]. Classifiers play important role in off-line handwriting recognition; they contribute greatly to the system performance. Suitable classifiers to the problems and the characteristics of feature extraction techniques are crucial as the recognition and accuracy results rely significantly on them. The research proposed here focuses on feature extraction techniques and artificial neural networks and support vector machines were chosen to be employed because they are appropriate to handle the proposed system problems. The employed classifiers are described as follows:

1 Artificial Neural Networks (ANNs)

Artificial Neural Networks or ANNs have been accepted that it is a powerful classifier for pattern classification problems. For off-line handwriting problems, ANNs are used to make the decision in classification phase. They use the most informative features which were obtained from employing a selected feature extraction technique(s) to determine which class each (testing) sample belongs to. There are numerous neural network structures available; however, the multilayer perceptron (MLP) and the radial basis function (RBF) are the two structures which are employed and have been significantly. Moreover, many researchers employed both of the structures in many applications and the results are reported. For this reason, the successfulness of the structures is well-known and accepted. However, many researchers have suggested from the results of their studies that the MLP networks generally work well on classification problems, while the RBF networks generally work well on function approximation problems.

Resilient propagation (Rprop) was proposed [187] to overcome the inherent disadvantages of pure gradient-descent. The Rprop performs a local adaptation of the weight-updates according to the behaviour of the error function, in the other words, it modifies the update-values for each weight according to the behaviour of the sequence of signs of the partial derivatives in each dimension of the weight-space. It is also considered as the best measured and algorithm regarding the convergence speed, accuracy and robustness with respect to training parameters [187], [188]. The Rprop, which is one of the feedforward artificial neural networks, was employed in this research to address the problem of magnitudes of the partial derivative effects when using the sigmoid function, and the MLP networks are known for their good performance for classification problems.

2 Support Vector Machines (SVMs)

Support Vector Machines (SVMs) are originally created for two-class classification problems and are known for their success classification algorithms and robustness. Rather than only employing the SVMs for two-class classification, new algorithms for multiclass problems have been proposed and employed successfully [189], [190], [191].

The binary SVMs can be combined in two ways which are One-Against-All (OAA) and One-Against-One (OAO) or pairwise. In OAO SVMs, for each pair of classes one binary SVM is constructed and trained to separate the data from the others. The study proposed here employed the OAO method as it was suggested to be working well with problems with a larger number of classes [189]. Kernels which are the elementary element shared by all kernel methods; they provide a general framework to represent data, and must satisfy some mathematic conditions. Radial Basis Function (RBF) kernel is one of the popular kernel functions which have been applied in many applications. The research proposed also employed the RBF kernel. The RBF kernel on two samples x and x', represents as feature vectors in some input space. The following kernel on $\mathcal{X} = \mathbb{R}^p$ called Gaussian RBF can be defined as:

$$K_G(x, x') = \exp(\frac{d(x,x')^2}{2\sigma^2}),$$

where σ is a parameter and d is the Euclidean distance [192]

E. Assessment Criterion

The criterion for assessment is important as it enables the system to mark the examination answers according to their quality. Human assessors generally consider the quality of the answers, and give partial marks to partially correct answers rather than mark the answers zero or one. This also applies to the SAAS. This type of system is clearly more usable and will benefit the students being examined as it allows partially correct answers to be marked accordingly. The assessment criterion was used in the marking phase. If the recognised word is a correct answer, the mark is given according to the marking scheme. For example, in the question "What does IT stand for?" the answer to this question contains 2 words which are "information" and "technology". If the answer contains any of those words, half marks will be awarded. The system's enhanced ability to mark more realistically (with partial marks) resulted from the inclusion of the marking criterion into the system.

F. Experimental Settings

For both ANN and SVM settings and structure, there were 12 outputs for the 12 answers. HP, CH, and HPCHC of each word belong to the same output. For example "Information" and "information" were classified as the same output.

For ANNs, the number of hidden units investigated during training was experimentally set from 20 to 120 hidden units. The number of iterations set for training increased from 500 to 10,000. The range of the number of hidden units was primarily designated based on experiments. The range was determined by observing the recognition rates obtained when the number of hidden units was increasing (by one hidden unit at a time). The lowest number of hidden units used in each experiment was low and was the number which allowed the recognition rate to rise. The maximum number of hidden units for each experiment was determined based on observation of recognition rates rising and lowering as the number of hidden units increased. For all experiments, resilient backpropagation algorithm (using MATLAB toolbox) was employed. Initially, the training process stopped when the designated maximum number of iterations was reached. The training process also stopped when the amount of time was exceeded, performance was minimised to the goal, and the performance gradient fell between minimum performance gradient.

Table 3.2: Feature extraction techniques employed in the proposed off-line Short Answer Assessment System (SAAS) and their corresponding gamma values

Feature Extraction Technique	Gamma Value
Modified Direction Feature (MDF)	0.0083
Gaussian Grid Feature (GGF)	0.0012
Water Reservoir Feature (WRF)	0.0026
Nine Loop Feature (LF – 9)	0.0083
Fifteen Loop Feature (LF – 15)	0.0052
Enhanced Water Reservoir, Loop, and GGF – Set A (EWRLGGF_A)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set B (EWRLGGF_B)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set C (EWRLGGF_C)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set D (EWRLGGF_D)	0.0007
Enhanced Gaussian Grid Feature – Set A (EGGF_A)	0.0011
Enhanced Gaussian Grid Feature – Set B (EGGF_B)	0.0011
Enhanced Gaussian Grid Feature – Set C (EGGF_C)	0.0011
Enhanced Gaussian Grid Feature – Set D (EGGF_D)	0.0011
Enhanced Modified Direction Feature – Set A (EMDF_A)	0.0078
Enhanced Modified Direction Feature – Set B (EMDF_B)	0.0078
Enhanced Modified Direction Feature – Set C (EMDF_C)	0.0073
Enhanced Modified Direction Feature – Set D (EMDF_D)	0.0072

The number of inputs has been described in Section II-B. For the SVM settings, a four-fold cross validation was performed to get consistent and meaningful results. The multi-class classification with radial basis function was used and the C parameter of the SVM was set to be 25. The C parameters were defined by employing cross-validation. A number of C values were used in the experiments (of each feature extraction technique) and the C value which yielded the best accuracy value was selected.

All experiments employing the SVM as the classifier utilised the Library for Support Vector Machines (LIBSVM) [194]. The gamma values employed in all experiments was the default value which is: $\gamma = \frac{1}{\text{Number of Features}}$. The gamma values applied for the employed feature extraction techniques can be seen in the table 3.2.

G. Classification Criteria

There were two minimum threshold values used as classification criteria. For threshold value ≥ 0.5 criterion, once each of the highest threshold valued outputs had been obtained, it was checked to evaluate if its value was more than or equal to 0.5. If it was, then it was recognised as the output and eligible for the marking process. If the output threshold value was lower than 0.5, it would be classified as an ambiguous word and would need to be manually marked. For the threshold value ≥ 0.0 criterion, the highest threshold output was recognised as the output as long as its threshold value was more than or equal to 0.0, and therefore was eligible for the marking process. No manual marking process took place as no minimum threshold applied.

The threshold value of 0.0 was set based on the fact that it was the lowest threshold value that allowed outputs with any of the threshold values to pass through to the recognition and marking/identification phases. The threshold value of 0.5 was set as its value was the middle value between 0 – 1; hence the threshold value of 0.5 was not too strict nor lenient for any outputs with reasonable threshold values (≥ 0.5) to pass through to the recognition and marking/identification phases. For both threshold values, normal training was performed on the datasets (e.g. for ANNs, the number of hidden units investigated was experimentally set from 20 up to 120 hidden units, incrementing by 1 at a time. The number of iterations set for training increased from 500 up to 10,000, incrementing by 500 at a time). Other threshold values such as 0.7 and 0.8 were also experimentally employed in the study; however, there was no substantial positive outcomes obtained. As a result, other threshold values employed were not included in the thesis.

H. SAAS Evaluations

There were three rates which were used to evaluate the SAAS, namely recognition, assessment accuracy and efficiency rates. Recognition rates were evaluated by using the datasets Type I and II. However, assessment accuracy and efficiency rates were evaluated by employing dataset Type II. The testing dataset included 132 samples for HP and CH SAAS, and 264 samples for HPCHC SAAS. Only the best recognition outcomes using each feature extraction technique were applied individually to the proposed systems. The assessment accuracy rates were then calculated. The assessment accuracy rate is the rate which indicates the accuracy of the proposed system when the recognised words matched the answers to each of the questions, whilst rejecting words classified as being ambiguous for manual assessment. The misrecognised rates were not included.

Assessment accuracy rates were obtained by: recognition rates of the correctly recognised answers + the rejection for manual assessment rates. The efficiency of the SAAS is the product of the recognition and assessment accuracy rates, it was calculated by:

$$(\text{Recognition Rate} \times \text{Assessment Accuracy Rate}) / 100$$

3.1.2 Student Identification System (SIS)

The Student Identification System (SIS) is a sub-system of an OFSASIS, which helps to identify students by recognising their handwritten components. These elements are usually found on examination answer sheets beside the student ID. The SIS benefits from off-line handwriting recognition techniques as well as the SAAS; however, there is no work has been reported on SIS. The SIS is used to identify students only and is not used to verify students. The main reason that the system employs student name components instead of the student ID is because, in the future, it is expected that a student identification and verification system could be developed. In another words, handwritten name components could be used to verify if the person who sat an examination is exactly the same person who 'owns' the name. Name components are used in SIS and not the student signature because generally students do not sign examination papers. The use of a signature, therefore, will not reflect the real-world application of the system, and for this reason was not used in the development of the proposed system.

There are three SISs proposed here which are 1) the English SIS (ESIS) 2) Thai SIS (TSIS) and 3) the Bilingual Student Identification System (BSIS). Whereas the ESIS and TSIS are only used for English and Thai language respectively, the BSIS is used for the two languages being Thai and English. The BSIS was created based on the fact that in some Thai examination papers, both Thai and English student names (written their names in Thai and English languages), together with the student number were written down to identify each student.

For the OFAAS, once the marking process from marking sub-system (SAAS) is completed, the report on each student's mark is automatically produced. This means that the mismatch between students' names and their marks can be reduced or eliminated. The system also produces a report for the human assessor indicating the names of the students who might have been absent from the examination. The system proposed in this research intends to identify students from their handwritten names, middle names and last names but with 'no intention' to verify whether the written names have been forged.

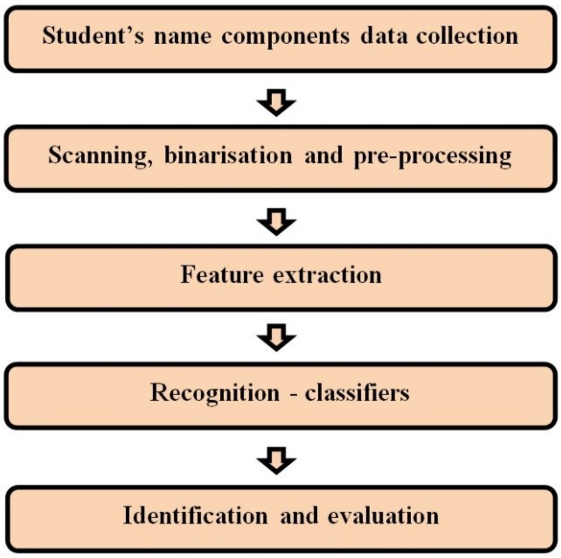


Figure 3.15: A block diagram illustrating the Student Identification System (SIS)

As can be seen in Figure 3.1, the proposed SIS which is one of the sub-systems has the function of identifying students from their name components. The proposed methodology includes data collection (students' name components), image processing, pre-processing, and utilises the proposed feature extraction techniques in conjunction with different classifiers, which are Support Vector Machines (SVMs) and Artificial Neural Networks (ANNs), in order to achieve the best results and for comparison purposes. The block diagram in Figure 3.15 illustrates the proposed system methodology and processes. The individual components of the system are discussed as follows.

A. Data Collection

There are no handwritten name components from examination papers publicly available; as a result, a data collection process was performed to create a custom dataset. The dataset collected for the proposed system is the first database of its type. The nature of the experiments proposed here, in comparison to other writer identification systems, may not be appropriate. The reason is because in other research work, the systems were attempting to identify writers from their handwritten words (which were not usually their names, and those same words were written by many writers). In the research proposed here, the recognition of words was based on one writer per name components. Although in some cases, the writers (students) may have had the same name components (such as their names, or last names), the system will identify the students by using the rest of their name components as well as by using other criteria (refer to Section H). This makes the systems in the literature and the SIS proposed here quite different. There are three datasets; one for each of the systems (ESIS, TSIS and BSIS). The ESIS, TSIS, and BSIS datasets are described as follows:

i. **The ESIS dataset:** handwritten names collected from one hundred writers were performed; the restricted number of one hundred volunteers was determined based on an assumption that for most classes, except for larger classes such as some lectures in universities, the number of students per class is less than one hundred. Therefore, for data collection, one hundred volunteers, both male and female of various nationalities, provided handwritten name components in order to be used for training and testing the proposed ESIS.

Ten name, and last name samples from each volunteer were collected. Some middle names and second last names were also provided by some volunteers. Altogether there were 2,040 (204 names × 10 samples of each name) samples obtained. All samples were written with minimum constraints (e.g. writing instruments and handwriting styles are not restricted within the given space). All the name components were written in English. Each volunteer wrote their name components one after another ten times. It must be noted that even though the volunteer wrote their name components one after the other, the intra-class differences were still found (Figure 3.16).

It is also noted that since the collection of handwritten full names for experimentation was less constrained, some volunteers chose to write their middle name as well as the second last name while most of the volunteers only wrote their first name and last name. Also because of the name component examples are from various nationalities, the name lengths also varied. For example, the name components collected from Chinese, Korean, and European volunteers were rather short, and they could be only one syllable (Li, Wang, Bu, Brown, and Jo, for instance). And whilst some nationalities' name components were short, some of the name components collected from other nationalities such as Thai and Columbian volunteers were particularly long, for example, Chalermwutthiphong, Pipatsattayanuwong and Leguizamon.

Many of the samples were collected from Thai volunteers. One of the Thai name characteristics was that the name components both first and last names may begin, end, or include some common words for name components such as "wat", "chai", "ya", "kit", and therefore can be quite confusing for the classification because of the common characteristic, which was shared by the volunteers. Some European name samples collected from volunteers were duplicated such "Michael", "Smith", "Judy". In conclusion, the dataset contains names with quite varied word lengths (from 1 – 8 syllables), some duplicated last and first names, and some of the name components contain common words within each component. Name component examples can be found in Figure 3.16. From those samples, different lengths of words were observed. Intra-class differences were found and name component duplication of different writers existed (name component of "Suwanwiwat" and "Akerlund" were written by four different writers – two of each).

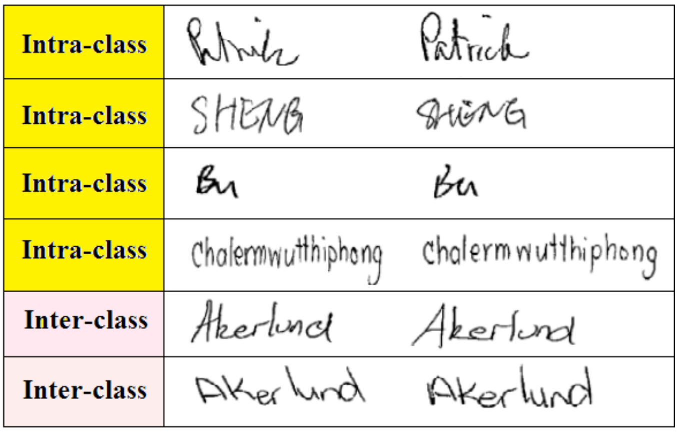


Figure 3.16: Name component examples. Intra-class differences, various lengths of name components, and duplication of name components written by different writers can be found

ii. **TSIS Dataset:** there is no publicly available dataset of Thai language handwritten name components from examination papers; as a result, a data collection process was performed to create a custom dataset. The dataset collected for the proposed system is the first database of its type in the Thai language. The restricted number of one hundred and three volunteers was determined based on an assumption that for most classes, the number of students per class is less than one hundred, except for larger classes such as some lectures in universities. Therefore, one hundred and three volunteers, both male and female, provided handwritten Thai language first and last names in order to be used for training and testing the proposed TSIS. In total, there were 2,060 (206 name components x 10 samples of each name component) samples obtained. All samples were written with minimum constraints (e.g. writing instruments and handwriting styles were not restricted within the given space). All the name components were written in Thai.

Each volunteer wrote their name components one after another ten times. Even though the volunteers wrote their name components one after the other, intra-class differences were still found. Some Thai name components are particularly long, for example, “สุวรรณวิวัฒน์”, “สวรรค์บรรจบริการ”, and “เอกลักษณานนท์”. One of the Thai name characteristics is that the name components both first and last names may begin, end, or include some common words for name components such as “wat”, “chai”, “ya”, “kit”, and therefore can be quite confusing for automatic classification because of the common characteristics, which were shared by the volunteers. In conclusion, the dataset contains names with quite varied word lengths (from 2 - 5 syllables which can be up to 14 characters), some duplicated last names, and some of the name components containing common words within each component. Name component examples can be found in Figure 3.18. From those samples, different lengths of words were observed. Intra-class differences were found and name component duplication of different writers existed (the name component of “สุวรรณวิวัฒน์” was written by two different writers).

iii. **BSIS Dataset:** the bilingual dataset contains Thai and English languages. The BSIS dataset was obtained from the existing ESIS and TSIS dataset; some of the modifications were made to the BSIS. The Thai dataset was obtained from the TSIS; the TSIS contains 2,060 (206 name components × 10 samples of each name component) sample. The English name component dataset was obtained from the ESIS dataset plus 6 additional English name components from 3 writers and the Thai name component dataset. To sum up, the bilingual dataset was increased to 4,120 (412 name components × 10 samples of each name component) samples. All samples were written with minimum constraints (e.g. writing instruments and handwriting styles were not restricted within the given space). Since the Thai name component dataset was collected at a different time to the English name component dataset, most the writers were not from the same group of people and therefore most of Thai and English names are not matched (did not belong to the same person).

Figure 3.17: Name component examples. Intra-class differences, various lengths of name components, and duplication of name components written by different writers can be found

In summary, the dataset contains names with quite varied word lengths (from 2–7 syllables which can be up to 18 characters), some duplicated last names, and some of the name components containing common words within each component. Name component examples can be found in Figure 3.17. From those samples, different lengths of words were observed. Intra-class differences were found and name component duplication of different writers existed (the name component of "สุวรรณวิวัฒน์" was written by two different writers).

B. Student Name Component Dataset

The ESIS, TSIS and BSIS student name component datasets are described as follows:

i. **The ESIS Dataset:** A total number of 2,040 word samples were used to facilitate training and testing; 80% of the samples of each name were used as the training dataset, and 20% were used as the testing dataset. Altogether there are 204 × 8 = 1,632 samples used for training and 204 × 2 = 408 samples used for testing.

ii. **The TSIS Dataset:** A total number of 2,060 word samples were used to facilitate ANN training and testing; 80% of the samples of each name were used as the training dataset, and 20% were used as the testing dataset. Altogether there are 206 x 8 = 1,648 samples used for training and 206 x 2 = 412 samples used for testing.

iii. **The BSIS Dataset:** A total number of 4,120 word samples were used for classifier training and testing; 80% of the samples of each name were used as the training set, and 20% were used as the testing set. Altogether there are 412 × 8 = 3,296 samples used for training and 412 × 2 = 824 samples used for testing.

C. Image Acquisition and Preprocessing

A scanner with 300 dpi resolution was used to perform image acquisition. The scanned images were stored in a grey-level format. After that, Otsu's thresholding algorithm was performed in order to obtain a binarised image [29]. Automatic line and word segmentation were performed using histogram projection. Line segmentation was performed first, and then word segmentation was performed in order to obtain each of the name components (first and last name). Salt and pepper noise removal (Algorithm 3.1) and filling (Algorithm 3.2) techniques were also applied. Skew normalisation [94] was performed on each name image. Slant correction was not performed on any of the images to preserve the unique characteristics of each student name.

For each image, boundary extraction was performed to isolate connected components. A boundary extraction algorithm was employed [61]. After that, loop as well as upper and lower contour extraction was performed. The upper and lower contour extraction was performed in order to find upper and lower reservoirs only and not to be used as features themselves; therefore, a skeletonisation approach was not considered in this research. Examples of preprocessing successive images are presented in Figure 3.18.

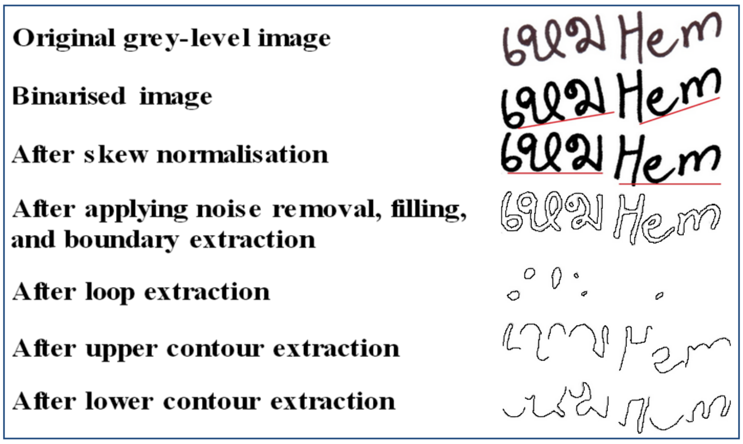


Figure 3.18: Example images obtained after each preprocessing step

D. Feature Extraction

Feature extraction techniques namely EWRGGF, EMDF, and EGGF, together with the original WRLGGF, MDF, and GGF were chosen in the experiment due to their ability to successfully extract important features from images, which have enabled accurate recognition rates to be attained in a number of applications [184], [82], [185], [86]. The water reservoir feature was used in the proposed BSIS here because both Thai and English have this property in their scripts (see Figure 3.19). Feature extraction techniques employed in the proposed Student Identification System (SIS) and their corresponding sizes can be seen in Table 3.3.

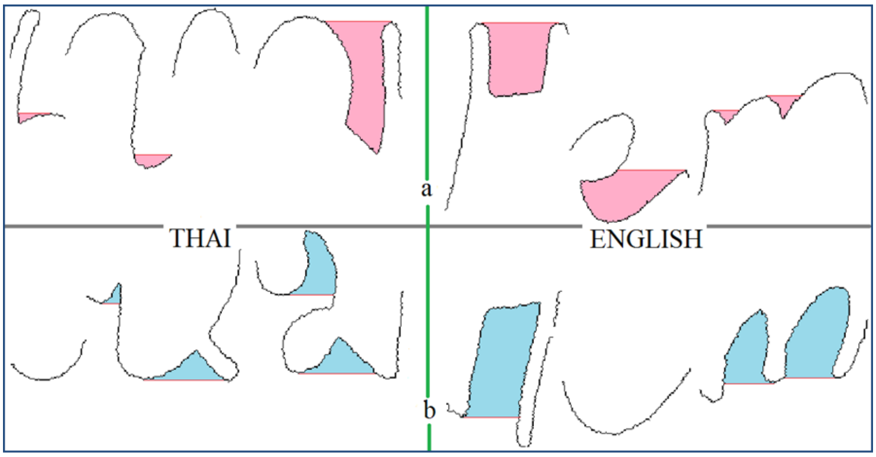


Figure 3.19: (a) Top WRs retrieved from the upper contour, and (b) bottom WRs retrieved from the lower contour. The full contour image can be seen in Figure 3.18

Table 3.3: Feature extraction techniques employed in the proposed Student Identification System (SIS) and their corresponding sizes

Feature Extraction Technique	Feature Vector Size
Modified Direction Feature (MDF)	121
Gaussian Grid Feature (GGF)	864
Water Reservoir Feature (WRF)	392
Nine Loop Feature (LF – 9)	120
Fifteen Loop Feature (LF – 15)	192
Water Reservoir, Loop, and GGF (WRLGGF)	1,448
Centre of Gravity Feature – Set A (Enhanced A)	8
Centre of Gravity Feature – Set B (Enhanced B)	8
Centre of Gravity Feature – Set C (Enhanced C)	16
Centre of Gravity Feature – Set D (Enhanced D)	18
Enhanced Water Reservoir, Loop, and GGF – Set A (EWRLGGF_A)	1,456
Enhanced Water Reservoir, Loop, and GGF – Set B (EWRLGGF_B)	1,456
Enhanced Water Reservoir, Loop, and GGF – Set C (EWRLGGF_C)	1,464
Enhanced Water Reservoir, Loop, and GGF – Set D (EWRLGGF_D)	1,466
Enhanced Gaussian Grid Feature – Set A (EGGF_A)	872
Enhanced Gaussian Grid Feature – Set B (EGGF_B)	872
Enhanced Gaussian Grid Feature – Set C (EGGF_C)	880
Enhanced Gaussian Grid Feature – Set D (EGGF_D)	882
Enhanced Modified Direction Feature – Set A (EMDF_A)	129
Enhanced Modified Direction Feature – Set B (EMDF_B)	129
Enhanced Modified Direction Feature – Set C (EMDF_C)	137
Enhanced Modified Direction Feature – Set D (EMDF_D)	139
Water Reservoir and Gaussian Grid Feature (WRGGF)	1,256
Loop and Gaussian Grid Feature(LGGF)	1,056
Water Reservoir, Loop and Modified Direction Feature (WRLMDF)	705
Water Reservoir and Modified Direction Feature (WRMDF)	513
Loop and Modified Direction Feature (LMDF)	313

The feature extraction techniques were applied separately and were not combined in any way. The features that were implemented in this research are EWRLGGF, EMDF, EGGF, water reservoir features, loop features, centre of mass feature, and the original WRLGGF, MDF, GGF. More details of each of the features employed in the proposed system can be found in Sub-Section 3.1.1.C.

Apart from the feature extraction techniques aforementioned, there are proposed techniques which were employed in the SISs. The proposed techniques are 1) Water Reservoir and Gaussian Grid Feature (WRGGF) extraction technique, 2) Loops and Gaussian Grid Feature (LGGF) Extraction Technique, 3) Water Reservoir, Loop, and Modified Direction Feature

Extraction Technique (WRLMDF). There are two loop vector sizes which employed loop feature extraction techniques in the SISs.

It should be noted that for the proposed hybrid feature extraction techniques, the features were not combined by family, but rather combined by the shared features between the two languages (English and Thai). That is both English and Thai languages have water reservoir, loop, and centre of gravity properties in their scripts. Hence when these features were combined with the modified direction feature or Gaussian grid feature, these hybrid feature extraction techniques increased the recognition/accuracy rates attained when employed on both of the languages. For the hybrid WRLGGF, the techniques were combined based on the hypothesis that by adding these properties, the efficiency of the original GGF will be increased. Details of the additional feature extraction techniques are described as follows:

1) Water Reservoir and Gaussian Grid Feature (WRGGF) Extraction Technique: the proposed WRGGF was developed based on two feature extraction techniques being Water Reservoir Feature (WRF) and GGF. WRF can be found in both Thai and English language, and are important character identifiers for the both languages, for example, ‘u’ from ‘o’ in English and ‘ก’ from ‘ม’ in Thai language. WRF was combined with GGF in order to extract the most salient information, so that the best recognition rates possible could be achieved. From full contour images, the process of top and bottom contour finding was undertaken. From the full boundary images, there were two images that were retrieved, being top contour and bottom contour image images (Figure 3.13 (a), (b), (c)). For the top and bottom contour, each reservoir was located, then reservoir feature extraction was applied on each reservoir and recorded in a feature vector. Finally, GGF was then applied on the full boundary contour to extract features from the image. The WRGGF vector size is 1,256 (392 from WRF + 864 from GGF).

2) Loop and Gaussian Grid Feature (LGGF) Extraction Technique: the proposed LGGF was developed based on two feature extraction techniques being Loop Feature (LF) and GGF. LF can be found in English, and is an important character identifier for both Thai and English language, for example, ‘ก’ from ‘ภ’ in Thai, and ‘d’ from ‘l’ for English language. LF was combined with GGF in order to extract the most salient information, so that the best recognition rates possible could be achieved. From full contour images, the process of loop finding was undertaken. From full boundary images, loop images were retrieved (Figure 3.13 (a) and (d)). Loop feature extraction was processed and features retrieved from each loop were recorded in the vector. Finally, GGF was then applied on the full boundary contour to extract features from the image. The LGGF vector size is 1,056 (192 from LF + 864 from GGF).

3) Water Reservoir, Loop, and Modified Direction Feature Extraction Technique (WRLMDF): The proposed WRLMDF was developed based on three feature extraction techniques being WRF, LFE and MDF. As WR and loop features can be found in both Thai and English, and are important character identifiers for the Thai language, the WRF and LFE were combined with MDF in order to extract the most salient information, so that the recognition rates could be enhanced. Vector size for WRLMDF is 705 (392 from WRF + 192 from LFE + 121 from MDF).

4) Water Reservoir and Modified Direction Feature (WRMDF) Extraction Technique: the proposed WRMDF was developed based on two feature extraction techniques being Water Reservoir Feature (WRF) and MDF. WRF can be found in both Thai and English language, and are important character identifiers for the both languages, for example, ‘q’ from ‘g’ in English and ‘ฤ’ from ‘ม’ in Thai language. Figure 3.20 illustrates an example of the use of the MDF and WRF extraction techniques combined. The technique would help classify letter “q” from “g” as there are more transitions from both top and bottom in the letter “g” more than “q”.

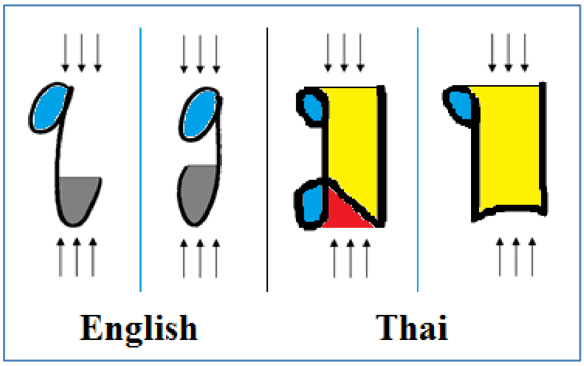


Figure 3.20: A hybrid MDF and water reservoir feature extraction technique

WRF was combined with MDF in order to extract the most salient information, so that the best recognition rates possible could be achieved. From full contour images, the process of top and bottom contour finding was undertaken. From full boundary images, there were two images that were retrieved, being top contour and bottom contour images (Figure 3.13 (a), (b), (c)). For the top and bottom contour, each reservoir was located, then reservoir feature extraction was applied on each reservoir and recorded in a feature vector. Finally, MDF was then applied on the full boundary contour to extract features from the image. The WRMDF vector size is 513 (392 from WRF + 121 from GGF).

5) Loop and Modified Direction Feature (LMDF) Extraction Technique: the proposed LMDF was developed based on two feature extraction techniques being Loop Feature (LF) and MDF. LF can be found in English language, and are important character identifiers for both Thai and English language, for example, ‘ฦ’ from ‘ฤ’ in Thai, and ‘d’ from ‘l’ for English language. LF was combined with MDF in order to extract the most salient information, so that the best recognition rates possible could be achieved.

From full contour images, the process of loop finding was undertaken. From full boundary images, loop images were retrieved (Figure 3.13 (a) and (d)). Loop feature extraction was processed and features retrieved from each loop were recorded in the vector. Finally, MDF was then applied on the full boundary contour to extract features from the image. The LMDF vector size is 313 (192 from LF + 121 from MDF).

6) Loop Feature (LF) Extraction Technique: As the nature of both Thai and English languages may contain loops in each character and because for the Thai language, the loops play an important role in distinguishing characters (refer to Section 3.2). For example ‘ฌ’ and ‘ญ’ in Thai, and ‘l’ and ‘d’ in English. The research proposed here also extracts features from loops found in images (see Figure 3.18). More details about LF can be found in Sub-section 3.1.1.C.4.

E. The Employment of Upper and Lower Contours, and Loops Images

Apart from employing full boundary images in feature extraction process, in this research, upper and lower contour and loops (the three images) of each full boundary image were also used in feature extraction process. The feature extraction techniques were performed on the three images without extracting from the full boundary images. This is to reduce the number of transitions in each direction when applied to the MDF. It must be noted that at the word level, the number of transitions is generally larger than at the character level; therefore, to simplify transitions in each direction, upper and lower contour images are used.

As illustrated in Figure 3.21, when finding transitions from background to foreground (1, 3, 5) and from foreground to background (2, 4), transitions of each direction can be more satisfactorily covered when applied on upper or lower contour images. Also, this was performed in order to investigate the efficiency of the feature extraction techniques to find out if less detailed images can yield as good as or better recognition rates when compared to the full boundary images. The employment of these three images was also used with other feature extraction techniques aside from the MDF.

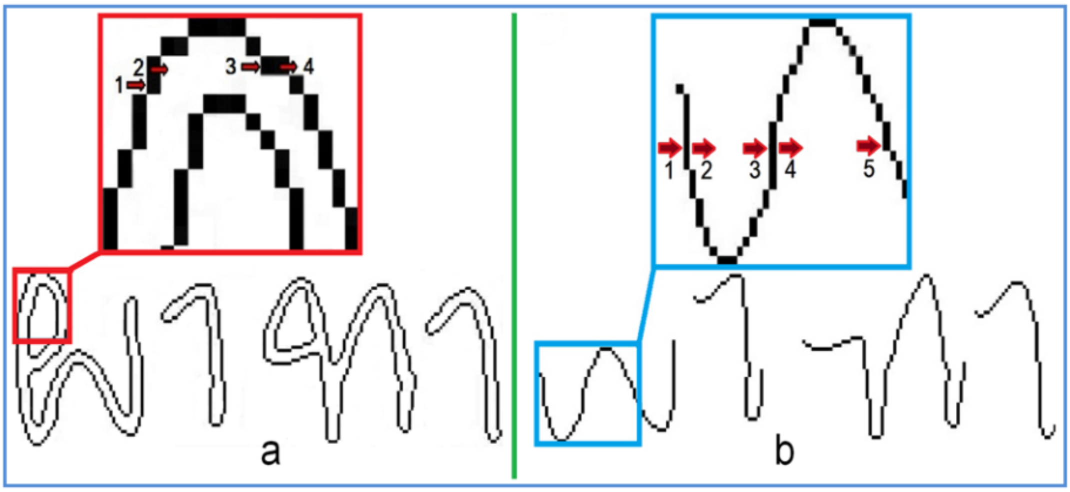


Figure 3.21: (a) An example of transitions on a full boundary image and (b) an example of transitions on a lower contour

F. Classification

The ANNs were trained with the resilient backpropagation algorithm, which was selected above all others to address the problem of magnitudes of the partial derivative effects when using the sigmoid function. For both MDF and GGF features, the neural networks were trained using 80% of the samples, and tested using 20% of the samples. For the experiments using SVM, libsvm [193] was employed in conjunction with the WEKA toolkit [193] and *libsvm* [194] was employed in conjunction with MATLAB. For training the SVMs, ten-fold cross validation was used across all 4,120 handwriting samples. The gamma values applied for the employed feature extraction techniques can be seen in the table 3.4 below:

Table 3.4: Feature extraction techniques employed in the proposed Student Identification System (SIS) and their corresponding gamma values

Feature Extraction Technique	Gamma Value
Modified Direction Feature (MDF)	0.0082
Gaussian Grid Feature (GGF)	0.0012
Water Reservoir, Loop, and GGF (WRLGGF)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set A (EWRLGGF_A)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set B (EWRLGGF_B)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set C (EWRLGGF_C)	0.0007
Enhanced Water Reservoir, Loop, and GGF – Set D (EWRLGGF_D)	0.0007
Enhanced Gaussian Grid Feature – Set A (EGGF_A)	0.0011
Enhanced Gaussian Grid Feature – Set B (EGGF_B)	0.0011
Enhanced Gaussian Grid Feature – Set C (EGGF_C)	0.0011
Enhanced Gaussian Grid Feature – Set D (EGGF_D)	0.0011
Enhanced Modified Direction Feature – Set A (EMDF_A)	0.0078
Enhanced Modified Direction Feature – Set B (EMDF_B)	0.0078
Enhanced Modified Direction Feature – Set C (EMDF_C)	0.0073
Enhanced Modified Direction Feature – Set D (EMDF_D)	0.0072
Water Reservoir and Gaussian Grid Feature (WRGGF)	0.0008
Loop and Gaussian Grid Feature(LGGF)	0.0009
Water Reservoir, Loop and Modified Direction Feature (WRLMDF)	0.0014
Water Reservoir and Modified Direction Feature (WRMDF)	0.0019
Loop and Modified Direction Feature (LMDF)	0.0032

G. Experiment Setting

The experiment settings of the ESIS, TSIS, and BSIS are described as follows:

i. **The ESIS:** For the ANN settings and structure, there were 204 outputs for the 204 first/middle/second last/last names. The duplicated name components from different writers, for example "Smith" from Jane Smith and "Smith" from John Smith were classified into 2 different outputs. However, in the recognition phase, Smith can be recognised as either Smith of John's output or of Jane's output. The SIS will identify who the name component belongs to. Classification into 2 different outputs was carried out because in the future it is believed that this will be useful in developing the SIS that can identify and verify students from their name components.

ii. **The TSIS:** for both ANN and SVM settings and structure, there were 206 outputs for the 206 first and last names. The duplicated name components from different writers, for example “สุวรรณวิวัฒน์” from “เหมพรรณ สุวรรณวิวัฒน์” and “สุวรรณวิวัฒน์” from “ธัญลักษณ์ สุวรรณวิวัฒน์” were classified into 2 different outputs. However, in the recognition phase, “สุวรรณวิวัฒน์” can be recognised as either “สุวรรณวิวัฒน์” of ธัญลักษณ์'s output or of ธัญลักษณ์'s output. The SIS will identify who the name component belongs to. Classification into 2 different outputs was carried out because in the future it is believed that this will be useful in developing the SIS that can identify and verify students from their name components.

For ANNs, the number of hidden units investigated during training was experimentally set from 30 up to 120 hidden units. The number of iterations set for training increased from 1000 up to 20000. All ANNs were trained with a learning rate of 0.l, and a momentum rate of 0.1. For the SVM settings, 10-fold cross validation was performed to get statistically meaningful results. The Gaussian kernel was used and the C parameter of the SVM was set to be 100.

iii. **The BSIS:** For both ANN and SVM settings and structures, there were 412 outputs for the 412 first and last names. The duplicated name components from different writers, for example “สุวรรณวิวัฒน์” from “บุญส่ง สุวรรณวิวัฒน์” and “สุวรรณวิวัฒน์” from “สมชาย สุวรรณวิวัฒน์” were classified into 2 different outputs. However, in the recognition phase, “สุวรรณวิวัฒน์” can be recognised as either “สุวรรณวิวัฒน์” of บุญส่ง's output or of สมชาย's output. The BSIS will identify who the name component belongs to. This is applied to both languages. Classification into 2 different outputs was carried out because in the future it is believed that this will be useful in developing the BSIS that can identify and verify students from their name components.

For ANNs, the number of hidden units investigated was experimentally set from 30 up to 130 hidden units, incrementing by 1 at a time. The number of iterations set for training increased from 1,000 up to 20,000, incrementing by 1,000 at a time. All ANNs were trained with MATLAB 2011a default parameters for resilient backpropagation. For the SVM experimental settings, ten-fold cross validation was performed to get statistically meaningful results. The Gaussian kernel was used and the C parameter of the SVM was set to be 100.

H. The SIS Evaluations

The evaluations of the ESIS, TSIS, and BSIS are described as follows:

i. **ESIS Evaluations:** Classification accuracy rates were evaluated by using the unseen testing dataset before being applied to the SIS. The unseen testing dataset included 408 samples (20% of the 2,040 samples). Only the best classification outcomes using each feature extraction technique were applied to the proposed system.

ii. **TSIS Evaluations:** Classification accuracy rates were evaluated by using the unseen testing dataset before being applied to the SIS. The unseen testing dataset included 412 samples (20% of the 2,060 samples). Only the best classification outcomes using each feature extraction technique from both classifiers (ANN and SVM) were applied to the proposed system.

iii. **BSIS Evaluations:** Classification accuracy rates were evaluated by using the unseen testing dataset before being applied to the BSIS. The unseen testing dataset included 824 samples (20% of the 4,120 samples). Only the best classification outcomes using each feature extraction technique from both classifiers (ANN and SVM) were applied to the proposed system.

I. The SIS criteria

After each of the name components were recognised, each SIS was used to map them against each student's name components. In order for the system to identify which student the name components belong to, it employs the following criteria:

- In the case that all the name components (first, middle, second last, last name) are recognised, the system will correlate the name component with the student who owns the name accordingly.
- In the case that only some of the name components are recognised, the system will check if the number of recognised student name components exceeds 50% of the total number of each student's name components. For example, if a student's name components include name, middle name, and last name and 67% (2/3) of the name components are recognised, then the system will identify the name components as that student.
- However, if the name components only include first and last name, and only one of these name components was recognised (50%), the system will check if there is any duplication with the name that was recognised. If there is, the system will check if another duplicated name has already been identified. If it has, then it will match the name components to the other person, which has not been identified with the same name component. For example, if there are students named "Jane Smith" and "John Smith", and only one name component of "Jane Smith" which is "Smith" was recognised; the system will check if "John Smith" was already identified. If "John Smith" has already been identified, the system will automatically identify the "Smith" as Jane Smith, otherwise it will reject this "Smith" for manual identification.
- If 50% of the name components were recognised, and there is no duplication of the name component with the other students, the system will match the name component to the student that owns it.
- If none of the name components has been recognised, the system will reject it for manual identification.
- In addition, for the BSIS system, both Thai and English names are available for each student name identification, even if some of the name components are misrecognised; the other name components of the other language are still available for identification. It

should be noted that a name component is classified as correctly Recognised (R) when it has the highest recognition rate from the classifier matching with the corresponding correct name of that name component; an Unrecognised (UR) name component is classified vice versa.

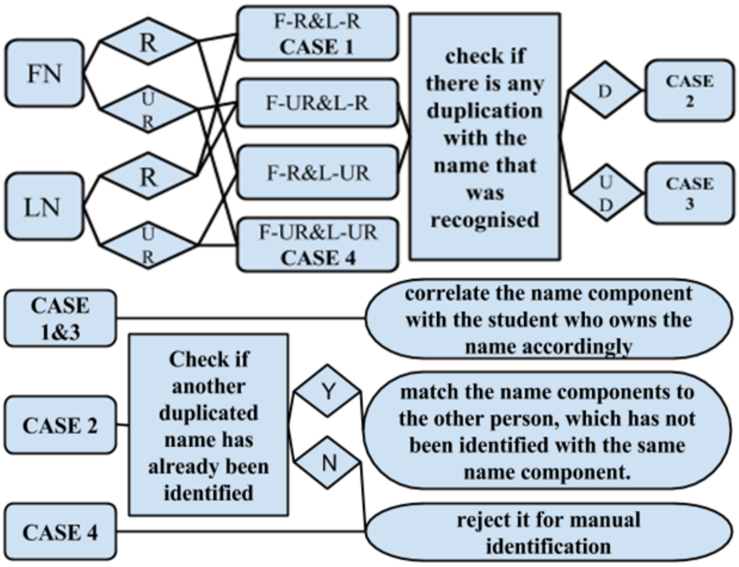


Figure 3.22: The SIS criteria

Figure 3.22 illustrates the SIS criteria decisions based on whether or not first (FN) or last (LN) name components are recognised (R) or unrecognised (UR), and if just one of the components is recognised, or whether there is any duplication (D) of FN or LN.

3.2 Thai Language

Thai language is the official language of Thailand; over 90% of its population used Thai in both writing and speaking language. Thai language has been developed since the mid-thirteen century. It was influenced by other countries including Cambodia (Khmer), India (Sanskrit and Pali) up until the late eighteen century [195]. Thai language has aspects which are similar and different to the English language. It is alphabetic as Thai language is written from top to bottom and from left to right. It is written in block letters (non-cursive); in other words, each of the characters is separated from its neighbours by a gap. However, in some cases when a Thai person writes quickly the characters or some of the characters can be connected. This can be seen in Figure 3.22 where the sentence "กินข้าวหรือยัง (KinKowRueYoung - Have you eaten yet?)" can be written in its usual block letters (Figure 3.23(a)) and its unusual connected (cursive-like) writing style (Figure 3.23(b)).

(a) (b)

Figure 3.23: Thai sentence in (a) block letters and (b) cursive-like letters

Thai language consists of 44 consonants, 18 vowels, 4 voice tones, and 3 special symbols, which can be seen in Table 3.5. Altogether, there are up to 69 characters (excluding Thai numerals) in the Thai language. Remarkably there is no space between words in Thai; however, there is a space between each sentence.

Table 3.5: Thai characters

Type	Type Members
Consonant	ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห อ ฮ
Vowels	ะ า อิ อี อึ อื อุ อู เ แ โ ไ ใ อั อ็ อำ ฤ ฦ (where อ can be any other consonant)
Tones	อ่ อ้ อ๊ อ๋ (where อ can be any other consonant)
Punctuation Marks	อ์ ฯ ๆ (where อ can be any other consonant)

Heads of Thai characters, which are small loops, play an important role because many of the characters look the same. Only the position or whether or not the head is present will tell what character it is. The heads of a character can be found in various locations. When classifying Thai characters using the character heads, three categories are obtained. These three categories are:

1) No-head character, for example, ‘ก’ for a consonant, ‘า’ for a vowel, and ‘่’ for a voice tone mark.
2) One-head character, for example, ‘ข’, ‘ผ’, ‘ล’, ‘อ’ for consonants, and ‘ไ’ for a vowel.
3) Two-head character, for example, ‘ณ’, ‘ฮ’, ‘ฬ’, ‘ฒ’ for consonants, and ‘ะ’ for a vowel.

An obvious example that shows the importance of the head is that these three different characters are only differentiated because of their heads: ‘ก’, ‘ภ’ and ‘ถ’. Loops (heads of characters) can be found in many positions, including upper-left part of a character (‘บ’), upper-right part of a character (‘ห’), middle part of a character (‘ฒ’), lower left part of a character (‘ม’), lower right part of character (‘น’).

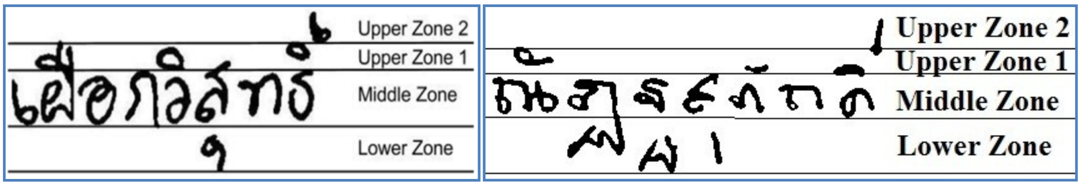


Figure 3.24: Thai sentence structure and its zones

There is no space between words for the Thai language, which makes it hard to segment words from a sentence. For example 'ตากลม' can mean sitting in the wind (ตาก ลม) or eyes wide open (ตากลม). However this is not a problem in the research proposed here as the names were recognised as a whole (whole word recognition). The last aspect to mention here is zoning. The Thai language structure can be classified into four main levels which are upper zone 2, upper zone 1, middle zone, and lower zone (see Figure. 3.24 for details).

Chapter 4

EXPERIMENTAL RESULTS

This chapter presents the results attained from employing the proposed methods and feature extraction techniques, which were described in Chapter 3, on the proposed short answer assessment with student identification systems. The results are shown in each sub-section according to each of the systems and their feature extraction techniques and classifiers employed.

This Chapter is organised as follows: Section 4.1 shows the experimental results of the proposed English student identification system. Section 4.2 presents the results of the proposed Thai student identification system. Details of the proposed bilingual student identification system are described in Section 4.3. The experiment results of the proposed off-line short answer automatic assessment system are presented in Section 4.4. Further analysis and discussion of the results presented in this chapter will be presented in Chapter 5.

4.1 Experimental Results of the Proposed English Student Identification System (ESIS)

The ESIS is a sub-system of an Off-Line Automatic Assessment System (OFLAAS), which helps to identify students by recognising their handwritten names, middle name, last names, and second last names (if applicable). These elements are usually found on examination answer sheets besides the student ID; the ESIS only identifies students and does not verify them. As the acronym ESIS suggests, the experiments were performed on English language datasets. Two sizes of the English datasets were used in the experiments, one dataset containing 2,040 samples and the other dataset containing 1,040 samples. Experiments employing each of the datasets are described as follows:

4.1.1 The 2,040 name component sample dataset

The first dataset used in this proposed system consists of 2,040 (204 names × 10 samples of each name) handwritten name components, which are first, middle, and last name, from a total number of 100 different writers. Training dataset contained 204 × 8 = 1,632 samples and 204 × 2 = 408 samples used for testing. After the samples were collected and had been scanned, binarised, noise removed, filled, and skew normalised, the image boundary extraction took place (Table 4.1). Both of the feature extraction techniques were performed on the boundary extracted images.

The experiments for ESIS were performed based on the employment of Gaussian Grid Feature (GGF) and Modified Direction Feature (MDF) extraction techniques. Feature vector sizes of the two techniques are quite different, while MDF vector size is 121, GGF vector size is

864. Originally, the MDF was created to extract direction and location information from handwritten characters at background - foreground transitions. Hence, the technique was developed to analyse information at the character level. However, the ESIS implemented the MDF to extract information from the whole word (each name component) image. Likewise, the recognition process utilised whole name information rather than recognising the handwriting at the character level.

Table 4.1: Example images after each preprocessing step

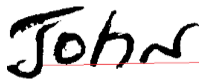
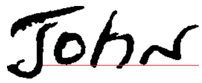
Process undertaken	Image
Original grey-level image	
Binarised image	
After skew normalisation	
After applying noise removal and filling	
After boundary extraction	

Table 4.2: Recognition rates attained employing GGF or MDF Feature Extraction Techniques in conjunction with the artificial neural network classifier

FET	RR (%)	Hidden Units	Iterations
GGF	98.28	72	12000
MDF	99.02	104	15000

The Artificial Neural Networks (ANNs), which were trained by 1,632 samples using either the GGF or the MDF extraction technique, were tested with the remaining 408 unseen samples. The best recognition rates, together with their settings are displayed in Table 4.2. These recognition rates are comparable (98.28% for GGF and 99.02% for MDF), although the numbers of hidden units and iteration settings are different (72 and 104 hidden units with 12,000 and 15,000 iterations, respectively). It is also noted that the training time required by the MDF feature is slightly shorter than the GGF due to its smaller vector.

The graph in Figure 4.1 shows the recognition rate comparison of the recognition rates obtained from employing the MDF and the GGF extraction techniques. A slightly better recognition rate of 0.74% was obtained by employing the MDF compared to the GGF (99.02% vs. 98.28%).

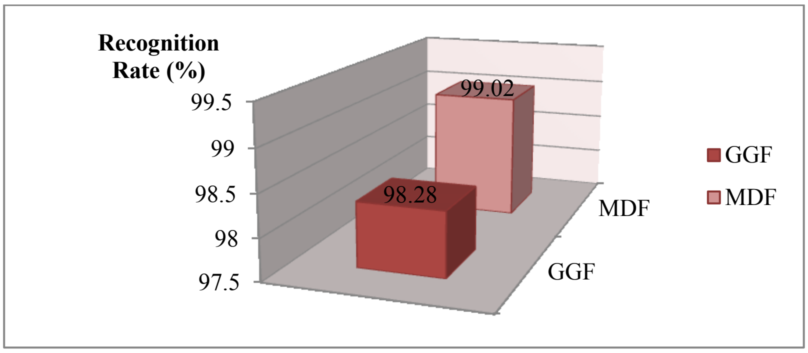


Figure 4.1: Recognition rates of the GGF and MDF extraction techniques

From the graphs in Figure 4.2 and Figure 4.3, it can be seen that even though only a slightly better recognition rate was obtained when employing the MDF as the feature extraction technique; the number of hidden units and the number of iterations are much higher when compared to the GGF. While the best MDF recognition rate employed 104 hidden units and 15,000 iterations, the best GGF recognition rate only employed 72 hidden units and 12,000 iterations. Therefore, the MDF employed 32 hidden units and 3,000 iterations more than the GGF.

With regards to the numbers of iterations, a graph illustrating recognition rates versus number of hidden units with iteration settings of 5,000, 10,000 and 15,000 for GGF and MDF classifiers can be found in Figure 4.4. From the graph it can be seen that the GGF results fluctuate more than MDF, also GGF seems to be more sensitive to the structure of the neural network (i.e. number of hidden units) than MDF. After 50 hidden units, the MDF results are rather stable. The summary of the recognition rates obtained employing GGF or MDF feature extraction techniques in conjunction with the artificial neural network classifier can be seen in Table 4.2.

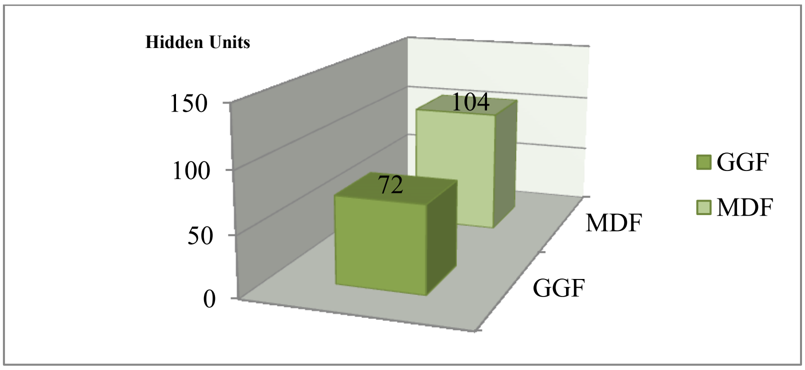


Figure 4.2: Number of hidden units for GGF and MDF extraction techniques

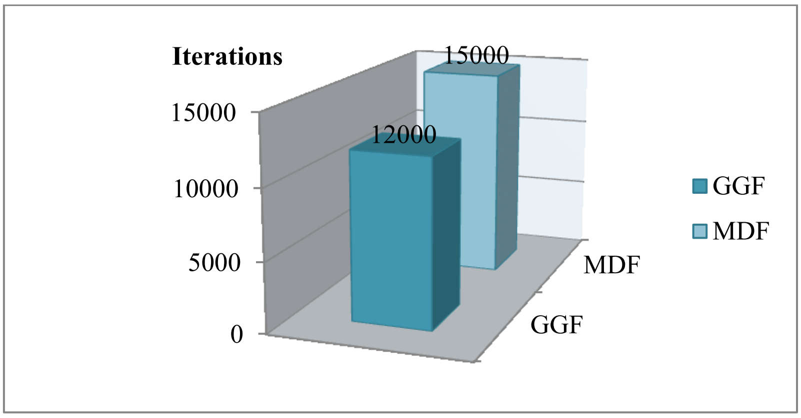


Figure 4.3: Number of iterations for GGF and MDF extraction techniques

Figure 4.4: Recognition rates versus number of hidden units with iteration settings of 5,000, 10,000 and 15,000 for GGF and MDF classifiers

After each of the name components were recognised, the SIS was used to map them against each student's name components. In order for the system to identify which student the name components belong to, it employs the criteria as described in chapter 3.1.2.I. One hundred percent accuracy was obtained when comparing the recognition rates of classifiers employing GGF or MDF extraction techniques and the identification accuracy of the ESIS with the human marker together with the above criteria. This 100% accuracy was attained, mainly because of the efficiency of the feature extraction techniques used, and the ability of the ESIS and its criteria to identify who the name components belong to, and its ability to reject some name components for manual identification.

4.1.2 The 1,040 name component sample dataset

The 1,040 sample dataset was employed for student identification sub-system of a complete automatic short answer assessment system with student identification. The name component samples were obtained from 52 examination papers which were written by 52 students. Altogether, there were 1,040 (104 name components (from 52 students) × 10 samples of each name components) samples obtained. The training set contained 832 samples and the testing set contained 208 samples. All preprocessing processes are the same as those in 4.1.1, except that the additional upper and lower contour, and loop extractions were also performed (see Table 4.3).

Table 4.3: Example images after each preprocessing step

Process undertaken	Image
Original grey-level image	
Binarised image	
After skew normalisation	
After applying noise removal, filling and boundary extraction	
After loop extraction	
After upper contour extraction	
After lower contour extraction	

The feature extraction techniques employed with this dataset are Enhanced hybrid feature extraction techniques called 1) Enhanced Water Reservoir, Loop and Gaussian Grid Feature (EWRLGGF), 2) Enhanced Gaussian Grid Feature (EGGF), 3) Enhanced Modified Direction Feature (EMDF), and the original techniques namely, Water Reservoir, Loop and Gaussian Grid Feature (WRLGGF), GGF, and MDF were utilised in the feature extraction technique performance investigation and comparison. The feature extraction techniques were applied separately and were not combined in any way. All details of the feature extraction techniques can be found in Chapter 3.1.1.C and 3.1.2.D.

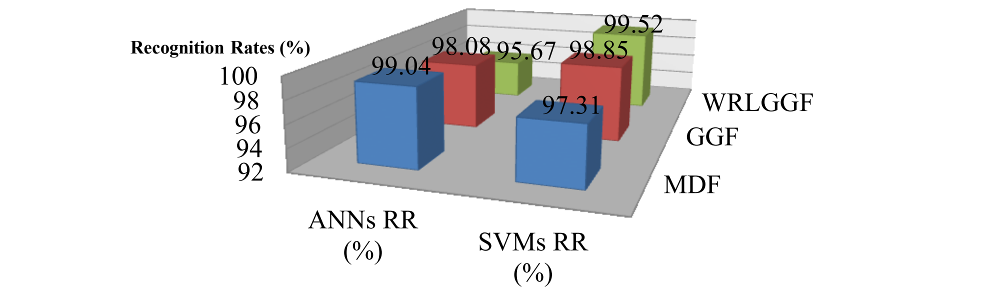
The ANNs were trained with the resilient backpropagation algorithm. For the ANNs, the number of hidden units investigated during training was experimentally set from 20 up to 120 hidden units. The number of iterations set for training increased from 500 up to 10,000. For the experiments using SVMs, *libsvm* [194] was employed in conjunction with MATLAB. Four-fold cross validation was used across all of the dataset. Multi-class classification with a radial basis function was employed, and the C parameter of the SVM was set to 35. For both ANN and SVM settings and structure, 104 outputs were set for the 104 first and last names.

The output of each name, which had the highest threshold regardless of its value, was recognised and was used in the identification process. The system evaluation is as described in Chapter 3.1.2.H. It can be seen from Table 4.4 that the best recognition rate was 99.52%. The best recognition rate was obtained by applying EMDF_B, EMDF_C, all of the EWRLGGFs, or the original WRLGGF (individually) as the feature vector and was 0.48% better when compared to the MDF of 99.04% using ANNs, and 0.67% better than the GGF using SVMs.

Table 4.4: Recognition Rate (RR) comparison between each Feature Extraction Technique (FET) applied to the dataset

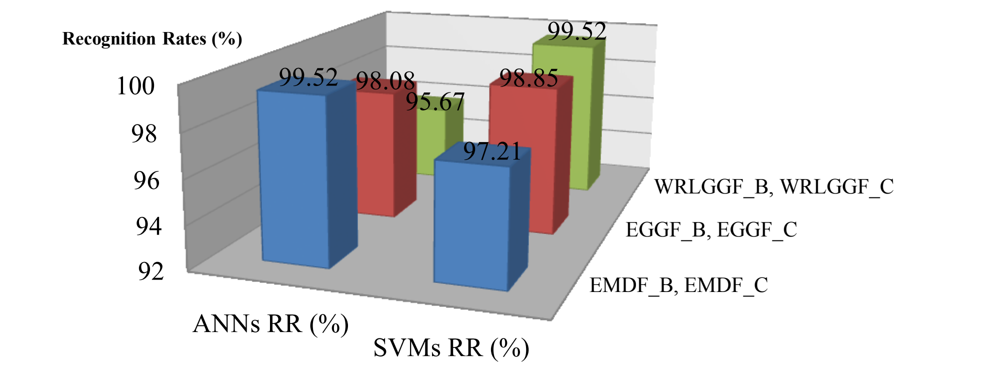
FET	RR (%)	
	ANNs	*SVMs*
MDF	99.04%	97.31%
EMDF_A	99.04%	97.12%
EMDF_B	**99.52%**	97.21%
EMDF_C	**99.52%**	97.21%
EMDF_D	99.04%	97.04%
GGF	98.08%	98.85%
EGGF_A	98.08%	98.85%
EGGF_B	98.08%	98.85%
EGGF_C	98.08%	98.85%
EGGF_D	98.08%	98.85%
WRLGGF	95.67%	**99.52%**
EWRLGGF_A	95.67%	**99.52%**
EWRLGGF_B	95.67%	**99.52%**
EWRLGGF_C	95.67%	**99.52%**
EWRLGGF_D	95.67%	**99.52%**

Figure 4.5 illustrates recognition rates of each of the original feature extraction techniques namely, the MDF, GGF, and WRLGGF. As can be seen, the best recognition rate obtained from employing the ANNs as the classifier is from utilising the MDF as feature extraction technique of 99.04%. This is 0.96% better than utilising the GGF, and is 3.37% better than employing the WRLGGF as feature extraction techniques. When employing the SVMs, however, the WRLGGF gave the best recognition rate of 99.52% which is the same best rate that the MDF attained when ANNs were utilised as the classifier. This is 0.67% better than the GGF and is 2.21% better than the MDF. It was observed that a slightly improved recognition rate was attained when employing the SVM with the GGF as the recognition rate improved by 0.88% (from 98.08% to 98.85%).



	ANNs RR (%)	SVMs RR (%)
MDF	99.04	97.31
GGF	98.08	98.85
WRLGGF	95.67	99.52

Figure 4.5: Comparison of recognition rates of the original feature extraction techniques namely, GGF, MDF and WRLGGF



	ANNs RR (%)	SVMs RR (%)
EMDF_B, EMDF_C	99.52	97.21
EGGF_B, EGGF_C	98.08	98.85
WRLGGF_B, WRLGGF_C	95.67	99.52

Figure 4.6: A graph comparing the best recognition rates of the enhanced GGF, MDF and WRLGGF

Figure 4.6 illustrates the best recognition rates obtained from employing the enhanced feature extraction techniques of each of the original feature extraction techniques (MDF, GGF, and WRLGGF) namely, EMDF_B, EMDF_C, EGGF_B, EGGF_C, EWRLGGF_B, and EWRLGGF_C. As can be seen, the pattern of the recognition rates is the same as for those obtained from the original feature extraction techniques (see Figure 4.5).

The best recognition rate obtained from employing the ANNs as the classifier is from utilising the EMDF_B and C as feature extraction techniques of 99.52%. This is 1.44% better than utilising the EGGF_B and C, and is 3.85% better than employing the EWRLGGF_B and C as feature extraction techniques. When employing the SVMs, however, the EWRLGGF_B and C gave the best recognition rate of 99.52% which is the same best rates that the EMDF_B and C attained when ANNs was utilised as the classifiers. This is 0.67% better than the GGF and is

2.31% better than the EMDF_B and C. A slightly improved recognition rate was attained when employing the SVM with the GGF as the recognition rate improved by 0.88% (from 98.08% to 98.85%). However, the EGGF_B and C had not been able improve the recognition rate of the original GGF.

The graph in Figure 4.7 displays the comparison of all feature extraction techniques employed utilising the ANNs as classifiers. For the MDF group, the EMDF_B and C yielded a slight improvement in recognition rate of 0.48% compared to the original MDF. However, the recognition rate is the same as for the original MDF when employed the EMDF_D. With the GGF group, the recognition rates are stable, and no changes are found when employed the enhanced GGF (EGGF_A, B, C, and D). The same recognition rate was obtained when employing each of the enhanced WRLGGF (EWRLGGF_A, B, C, and D) to ANNs; the recognition rate of 99.08% was obtained and remained unchanged with the original WRLGGF and the EWRLGGF techniques.

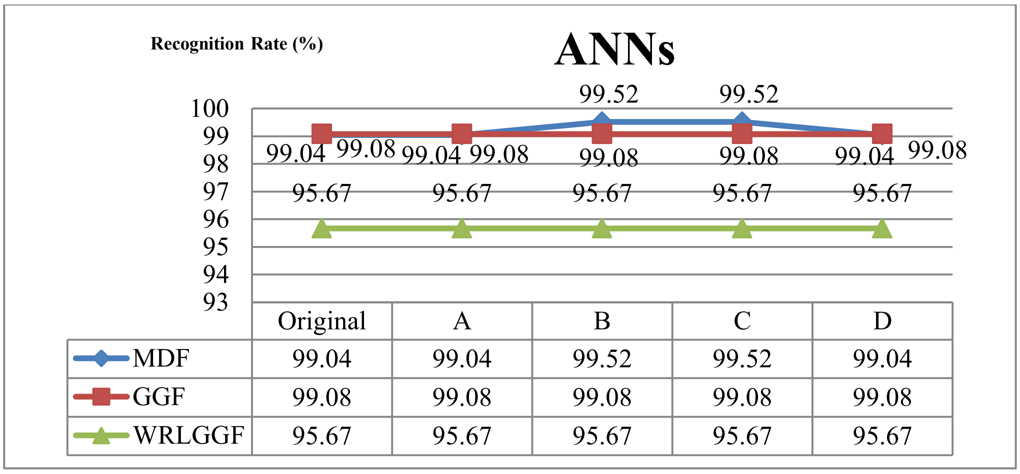
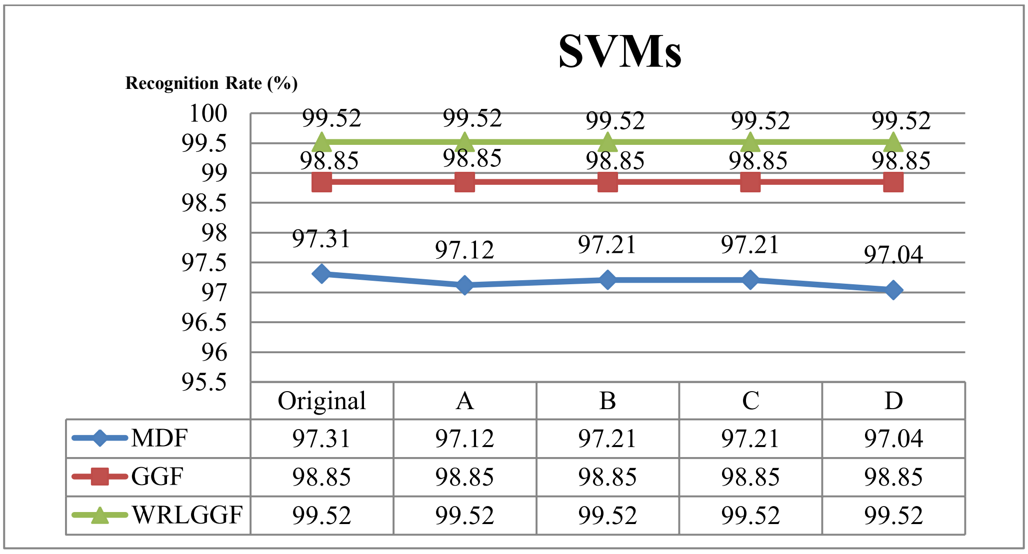


Figure 4.7: A graph comparing recognition rates of each feature extraction technique employing ANNs as the classifier

Figure 4.8 illustrates the recognition rates obtained when utilising SVMs as classifiers. It can be seen that the recognition rates obtained from employing the GGF and WRLGGF feature extraction groups have no changes in their rates. The original GGF, WRLGGF, and their enhanced feature extraction techniques yielded no improvements in obtaining recognition rates. The recognition rate of 99.52%, which is the highest recognition rate in this experiment, was obtained when employing the WRLGGF and its enhanced techniques (EWRLGGF_A, B, C, and D). The recognition rate of 98.85% was obtained from the GGF and its enhanced techniques (GGF_A, B, C, and D).



	Original	A	B	C	D
MDF	97.31	97.12	97.21	97.21	97.04
GGF	98.85	98.85	98.85	98.85	98.85
WRLGGF	99.52	99.52	99.52	99.52	99.52

Figure 4.8: A graph comparing recognition rates of each feature extraction technique employing SVMs as the classifier

4.2 Experimental Results of the Proposed Thai Student Identification System (TSIS)

The TSIS is a sub-system of an Off-Line Automatic Assessment System (OFLAAS), which helps to identify students by recognising their handwritten first and last names. These elements are usually found on examination answer sheets besides the student ID; same as the ESIS, the TSIS only identifies students and does not verify them. As the acronym TSIS, suggests the experiments were performed on a Thai language dataset containing 2,060 (206 name components x 10 samples of each name component) samples.

All samples were written with minimum constraints e.g. writing instruments and handwriting styles were not restricted within the given space. All the name components were written in Thai. Image acquisition was performed by using a scanner. The resolution of the images was 300 dpi; the scanned images were stored in a grey-level format. Binarisation was then performed on each image. Automatic line and word segmentations were performed using histogram projection. Line segmentation was performed first, and then word segmentation was performed in order to obtain each of the name components (first and last name). Salt and pepper noise removal and filling techniques were also performed. Skew normalisation was applied to each name image. To preserve the unique characteristics of each student name, slant correction was not performed on any of the images. The preprocessing techniques employed details can be found in Chapter 3.1.1.B.

For each image, boundary extraction was performed to isolate connected components. After that, loop as well as upper and lower contour extractions were performed. The feature extraction techniques which were selected in the proposed system are the MDF and GGF. After the feature vectors are generated by employing each technique, the features were then applied to the ANNs and SVMs for training, and testing for the recognition/identification process.

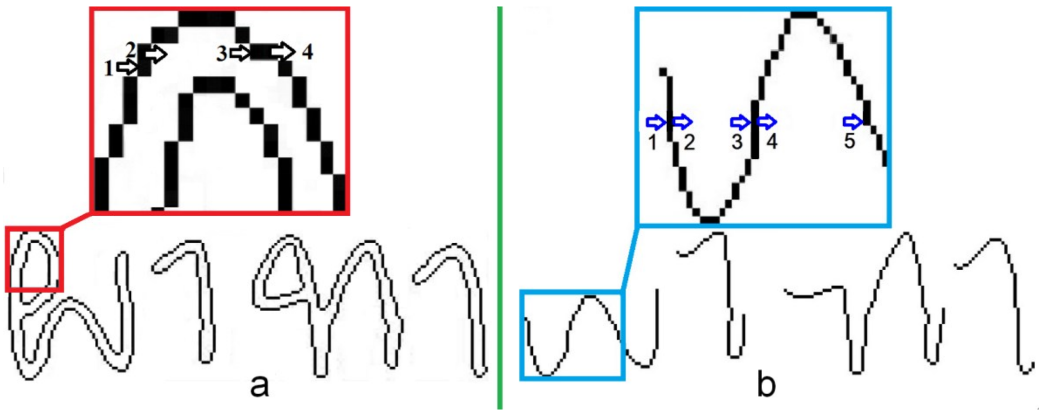


Figure 4.9 (a) An example of transitions on a full boundary image and (b) an example of transitions on a lower contour

Upper and lower contour images were employed in the feature extraction process to investigate if the feature extraction techniques employed could work efficiently compared to the full contour images. This is particularly with the MDF as it was initially created to be used at the character level. In this proposed research, the upper and lower contours extracted from full boundary images were also employed. This is to reduce the number of transitions in each direction when applied to the MDF. It must be noted that at the word level, the number of transitions is generally larger than at the character level; therefore, to simplify transitions in each direction, upper and lower contour images were used. As illustrated in Figure 4.9 (a) and (b), when finding transitions from background to foreground (1, 3, 5) and from foreground to background (2, 4), transitions of each direction can be more satisfactorily covered when applied on upper or lower contour images.

For each of the experiments, each feature extraction technique (MDF and GGF) was employed to extract features from 1) full contour images and 2) the Three Images (TI) being upper contour, lower contour, and loop images.

A total number of 2,060 word samples were used for both GGF and MDF extraction techniques. The ANNs were trained with the resilient backpropagation algorithm, which was selected above all others to address the problem of magnitudes of the partial derivative effects when using the sigmoid function. The neural networks were trained using 206 × 8 = 1,648 samples, and tested using 206 × 2 = 412 samples for the experiments using SVM, *libsvm* [194] was employed in conjunction with the WEKA toolkit [193]. For training the SVMs, ten-fold cross validation was used across all 2,060 handwriting samples.

There were 206 outputs for the 206 first and last names. The duplicated name components from different writers for both ANN and SVM settings and structure, for example "สุวรรณวิวัฒน์" from "เหมพรรณ สุวรรณวิวัฒน์" and "สุวรรณวิวัฒน์" from "ธนวัฒน์ สุวรรณวิวัฒน์" were classified into 2 different outputs. However, in the recognition phase, "สุวรรณวิวัฒน์" can be recognised as either "สุวรรณวิวัฒน์" of เหมพรรณ's output or of ธนวัฒน์'s output. The TSIS will identify who the name component belongs to. Classification into 2 different outputs was carried out because, in the future, it is believed that this will be useful in developing the SIS that can identify and verify students from their name components.

For ANNs, the number of hidden units investigated during training was experimentally set from 30 up to 120 hidden units. The number of iterations set for training increased from 1,000 up to 20,000. For the SVM settings, ten-fold cross validation was performed to get statistically meaningful results. The Gaussian kernel was used and the C parameter of the SVM was set to be 100. Experiments employing the dataset are described as follows:

From Figure 4.10, when utilising ANNs as the classifiers, it can be seen that for MDF technique, the best recognition rate of 99.27% was attained when employing TI (upper contour, lower contour, and loop images) compared to the recognition rate of 99.03% of the features which were extracted from the full contour images. However, for the GGF, the recognition rate was dropped dramatically by 16.75% from 97.33% to 80.58%, when features were extracted from TI rather than the full contour images. When observed the number of hidden units and iterations, it can be seen in Figure 4.11 and 4.12 that even though only a slight improvement of 0.24% was obtained when employing TI rather than full contour images, the number of hidden units employed was 70; that is, 10 hidden units more than the number which were used for the full contour. Nevertheless, the number of iterations employed was only compared to the larger number of 15,000 iterations when extracting the features from the full contour images.

A number of 74 hidden units and 4,000 iterations were employed by the GGF on full contour images and yielded 97.33%, compared to the recognition rate of 80.58 % when employing 58 hidden units and 11,000 iterations on TI. The full contour images required more hidden units but less number of iterations to obtain a better recognition rate compared to the TI.

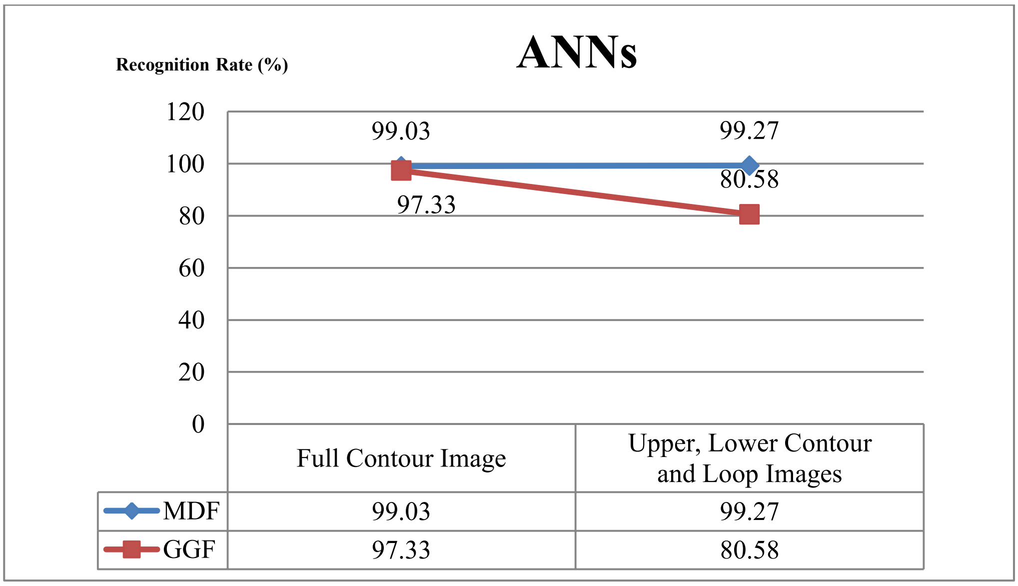


Figure 4.10: A graph comparing recognition rates of each feature extraction technique employing the ANNs as the classifier

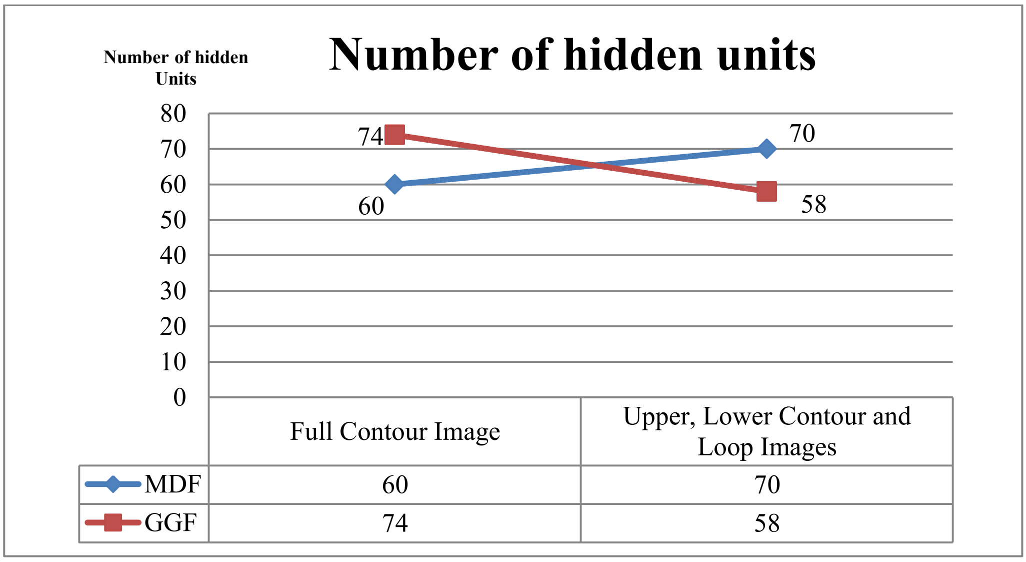


Figure 4.11: A graph illustrating the number of hidden units of each feature extraction technique

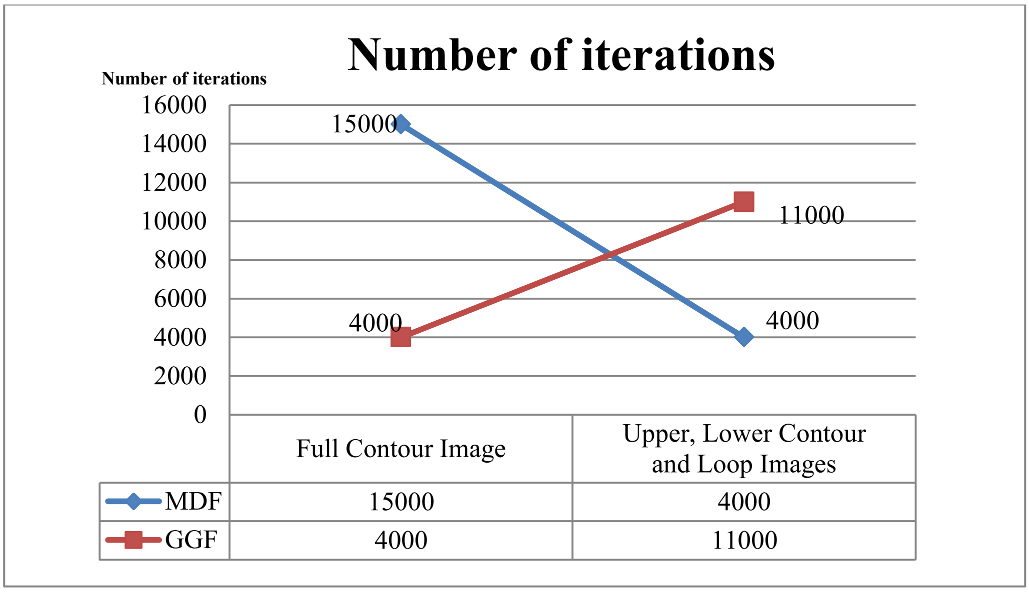
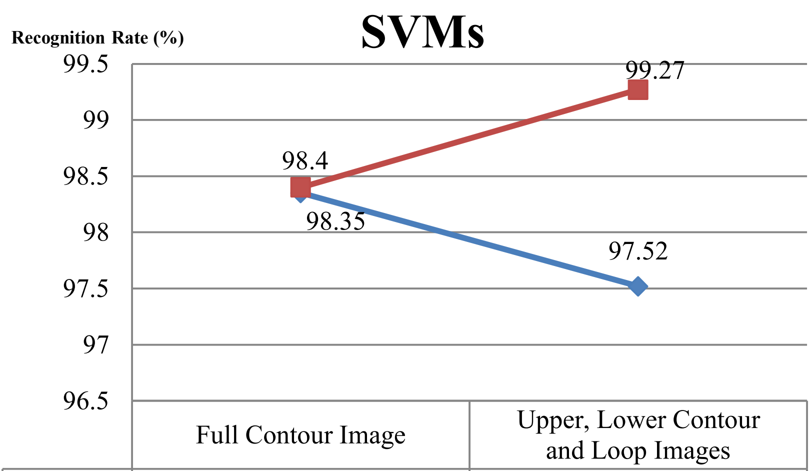


Figure 4.12: A graph illustrating the number of iterations of each feature extraction technique

Conversely when employing the SVMs rather than the ANNs as classifiers, it can be seen from Figure 4.13 that the recognition rates obtained from employing GGF on the TI was raised from 98.40% to 99.27%. However, when employing the MDF with SVMs, the recognition rate dropped from 98.35% when the feature was extracted from the full contour to 97.52% when employing TI; or that is 0.83% lower than the rate obtained when the features were extracted from the full contour image.



	Full Contour Image	Upper, Lower Contour and Loop Images
MDF	98.35	97.52
GGF	98.4	99.27

Figure 4.13: A graph comparing the recognition rate of each feature extraction technique employing the SVMs as the classifier

4.3 Experimental Results of the Proposed Bilingual Student Identification System (BSIS)

The BSIS identifies students from their name components (first and last name). Once the marking process is completed, the full report containing a list of students who attended the examination along with the marks they achieved is produced. The BSIS process begins with the data collection of the students' name components. The scanning process is used to transform raw data into digitised patterns. Binarisation and preprocessing, including line and word segmentation, noise removal, filling and skew correction, upper and lower contour, and loop extraction were then applied to the images. After the feature vectors are generated by applying each technique to the relevant digital images, the features are then input to the ANNs and SVMs for training, and testing for the recognition/identification process. Two sizes of the bilingual datasets were used in the experiments, one dataset containing 4,120 samples and the other dataset containing 7,880 samples. Experiments employing each of the datasets are described as follows:

4.3.1 The 4,120 name component sample dataset

The first dataset used in this proposed system consists of 4,120 (412 names × 10 samples of each name) handwritten name components (2,060 samples being Thai and the other 2,060 samples being English) which are first, middle (if applicable for English name components), and last names, from a total number of 206 different writers. After the samples were collected, scanned, binarised, noise removed, filled, and skews normalised, the image boundary, loop, upper contour, and lower contour extractions then took place (see 3.1.2.B).

The feature extraction techniques which were selected for investigation are the WRLGGF, WRLMDF, WRGGF, WRMDF, LGGF, LMDF, the original GGF and MDF. Details of these feature extraction techniques can be found in Chapter 3.1.2.C. For all feature extraction techniques, a total number of 4,120 word samples were used for classifier training and testing. The ANNs were trained with the resilient backpropagation algorithm. For both MDF and GGF features, the neural networks were trained using 412 × 8 = 3,296 samples, and tested using 412 × 2 = 824 samples. The number of hidden units investigated was experimentally set from 30 up to 130 hidden units, incrementing by 1 at a time. The number of iterations set for training increased from 1,000 up to 20,000, incrementing by 1,000 at a time.

For the experiments using SVM, *libsvm* [194] was employed in conjunction with the WEKA toolkit [193]. For training the SVMs, ten-fold cross validation was used across all 4,120 handwriting samples. The ten-fold cross validation was performed to get statistically meaningful results. The Gaussian kernel was used and the C parameter of the SVM was set to be 100.

For both ANN and SVM settings and structures, there were 412 outputs for the 412 first and last names. Classification accuracy rates were evaluated by using the unseen testing dataset before being applied to the BSIS. The unseen testing dataset included 824 samples (20% of the 4,120 samples). Only the best classification outcomes using each feature extraction technique from both classifiers (ANN and SVM) were applied to the proposed system. ANNs and SVMs were trained and tested using the five extraction techniques individually.

The recognition rates obtained from the ANN and SVM classifiers trained using MDF, GGF, WRGGF, LGGF or WRLGGF feature extraction techniques, which are described below. The best recognition rates, together with their settings, are presented in Table 4.5.

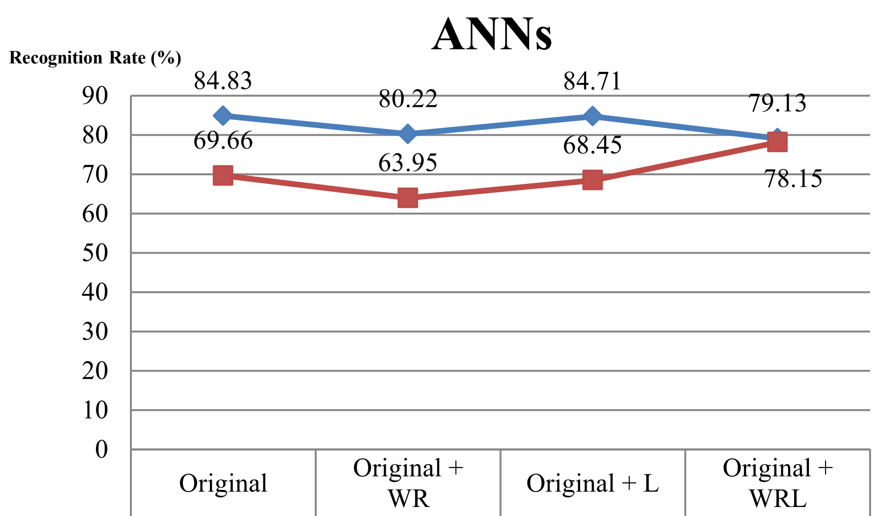
Table 4.5: Recognition rates attained employing each feature extraction technique in conjunction with the Artificial Neural Networks and the Support Vector Machines classifiers

	Feature Extraction Techniques							
Classifier	***MDF***	***WRMDF***	***LMDF***	***WRLMDF***	***GGF***	***WRGGF***	***LGGF***	***WRLGGF***
ANNs (%)	84.83	80.22	84.71	79.13	69.66	63.95	68.45	78.15
HU	127	126	118	109	120	111	97	121
Iterations	2000	3000	3000	3000	3000	2000	4000	2000
SVMs (%)	96.63	93.62	94.59	95.40	98.59	99.17	98.81	**99.25**

The best result of 99.25% was obtained using the WRLGGF with SVM setting. This result is 0.66% better than the GGF feature, which was originally designed for the signature verification problem. The improved result obtained when WR and loop features were extracted also from the full boundary/contour images was a result of more important information being extracted and therefore assisted in the recognition process. It was also observed that incorporating the loop features into the GGF slightly helped to improve the recognition rates from 99.17% (WRGGF) to 99.25% (WRLGGF), and from 98.59% (GGF) to 98.81% (LGGF).

The highest recognition rate retrieved from applying WRLMDF which is 95.40% however, did not outperform the recognition rates received from WRLGGF (99.25%) nor the original MDF of 96.63% (all best rates obtained using the SVM). It was found that both the loop and the WR features degraded recognition rates when compared to the features extracted from original MDF. This may be caused by additional WR and loop features having been too complex in the case of some classes and have introduced uncertainty in terms of additional, non-salient information.

In Figure 4.14, in which all experiments employed the ANNs as the classifier, it can be seen that for the MDF group, the best recognition rate of 84.83% was obtained by employing the original MDF. The recognition rates of the WRLMDF, WRMDF, and LMDF could not outperform the original MDF. The WRLMDF yielded 79.13% which is 5.7% lower than the original MDF. On the other hand, for the GGF group, the best recognition rate was obtained by employing the WRLGGF. The best recognition rate of 78.15% was obtained compared to the recognition rate of 69.66%; that is, an improvement of 8.49% was achieved.



	Original	Original + WR	Original + L	Original + WRL
MDF	84.83	80.22	84.71	79.13
GGF	69.66	63.95	68.45	78.15

Figure 4.14: Recognition rates for each feature extraction technique employing the ANNs as the classifier

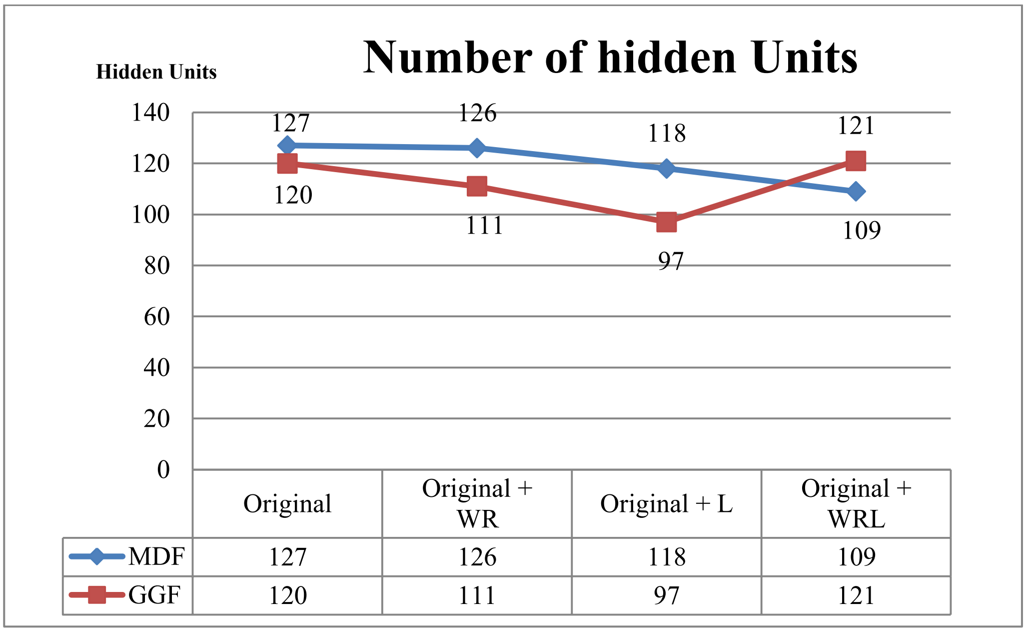


Figure 4.15: Number of hidden units of each feature extraction technique

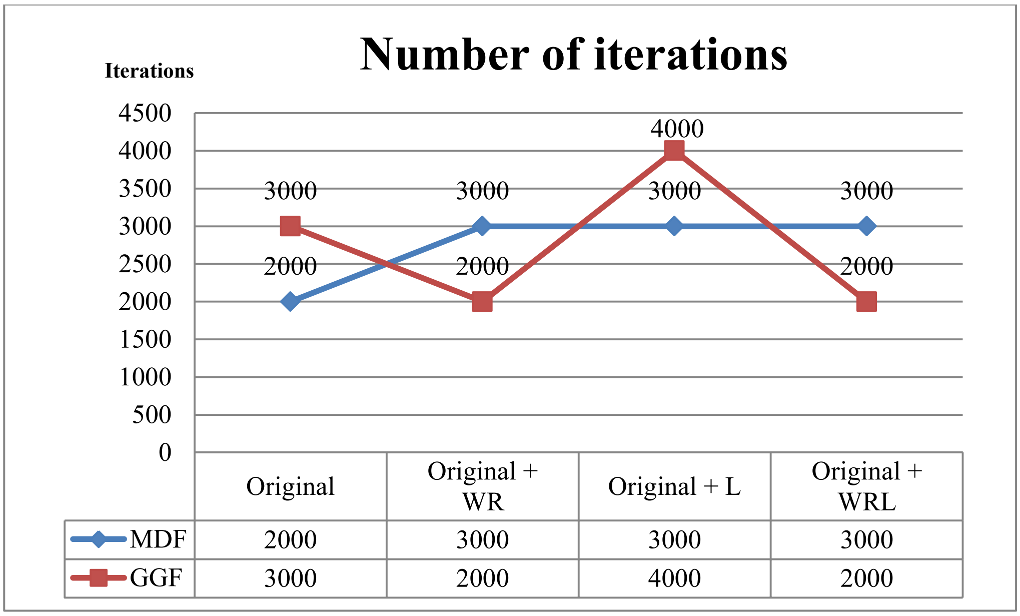


Figure 4.16: Number of iterations of each feature extraction technique

As mentioned earlier, an improvement in recognition rate of 8.49% was attained when comparing WRLGGF to the original GGF; from the graphs in Figure 4.15 and 4.16, it can be observed although quite the improvement occurred, the number of hidden units and the number of iterations of the WRLGGF are lower than those of the GGF's, the number of hidden units of WRLGGF being 121, and the number of iterations being 2,000. The number of hidden unit of 121 is one more than the original GGF of 120 and the number of iterations employed was 2,000 which was 1,000 iterations less than the original GGF of 3,000; as the result, the WRLGGF can be considered efficient and stable compared to its original counterpart.

It can be seen that for the modified MDF (WRMDF, LMDF, and WRLMDF), the number of iteration of 3,000 gave the best recognition rates (80.22%, 84.71%, and 79.13%); however, the rates obtained could not outperform the original MDF (84.83%). It is also observed that the best recognition setting of the MDF only employed 2,000 iterations, however, employed the most number of hidden units of 127 compared to the numbers of 126, 118, and 109 of the WRMDF, LMDF, and WRLMDF, respectively.

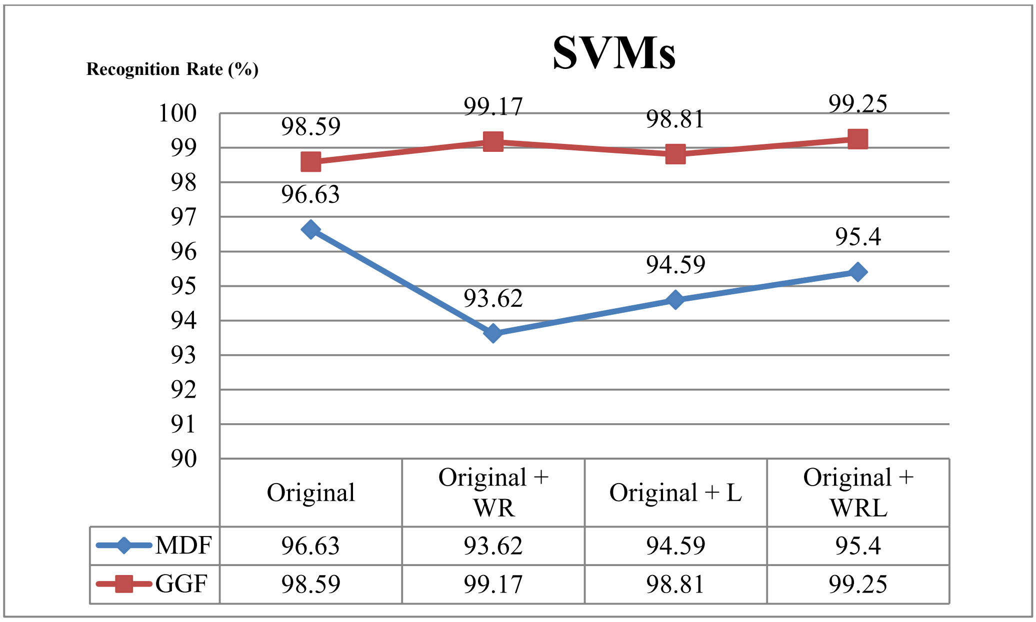


Figure 4.17: Recognition rates for each feature extraction technique employing the SVMs as the classifier

Figure 4.17 illustrates the recognition rates attained from employing SVMs as classifiers. It can be observed that the recognition rates obtained from employing GGF feature group are relatively stable. The recognition rates of 98.59%, 99.17%, 98.81, and 99.25% from the original GGF, WRGGF, LGGF, and WRLGGF were obtained. The best improvement of 0.66% was attained when employing the WRLGGF compared to the original GGF. The best recognition of 99.25% from the WRLGGF is also higher than the best recognition rate of 96.63 of the MDF (2.62%), which is the highest rate obtained from the MDF feature group.

The recognition rates employing the original MDF, WRMDF, LMDF, and WRLMDF are 96.63%, 96.62%, 94.59%, and 95.4% respectively. The modified WRMDF, LMDF, and WRLMDF could not outperform the original MDF when employing the SVMs as the classifier. This is similar to the results when employing the ANNs as the classifier, however, the recognition rates were more stable; the recognition rates also dropped less (84.83% dropped to 80.22%, 84.71% , and 79.3%) as those which were obtained from the SVMs (96.63% dropped to 93.62%, 94.59%, and 95.4%).

For all feature extraction techniques, the training times required when applying the MDF, WRMDF, LMDF and WRLMDF were shorter than for WRLGGF, WRGGF, LGGF, and GGF due to its smaller feature vector size. Also as can be seen, better recognition rates were obtained for when using SVM as the classifier. After each of the name components were recognised, the BSIS was used to map them against each student's name components. In order for the system to identify the student to whom the name components belong, it employs the BSIS criteria (refer to Chapter 3.1.2.I.).

One hundred percent accuracy was attained when comparing the recognition rates of classifiers employing WRLGGF or MDF extraction techniques and the identification accuracy of the BSIS with the human marker together with the above criteria. The 100% accuracy was obtained, mainly because of the efficiency of the feature extraction techniques used, combined with the ability of the BSIS and its criteria to identify to whom the name components belong, and its ability to reject some name components for manual identification. Another advantage is that because both Thai and English names are available for each student name identification, even if some of the name components are misrecognised, the other name components of the other language are still available for identification.

4.3.2 The 7,880 name component sample dataset

The second dataset which was used in experiments contained 7,880 samples which comprised Thai and English datasets combined; details of each dataset is as follows. The Thai dataset contained 3,940 samples of which 1,880 were newly collected. The other 2,060 samples were obtained from the Thai dataset in sub-section 4.3.1. For the English dataset, 1,880 newly collected English name components were combined with 2,060 samples from the English dataset in sub-section 4.3.1. In total, there were 3,940 samples in the English dataset. The total dataset (both Thai and English) was preprocessed the same way as those in sub-section 4.3.1. From the full boundary images, upper and lower contours, as well as loop images were extracted in order to be used in the feature extraction phase, as in sub-section 4.3.1. Feature extraction techniques which were used in the experiments were MDF, GGF, WRLMDF, and WRLGGF (refer to chapter 3.1.2.C); each of the feature extraction techniques were performed on 1) the full boundary contour and 2) on the Three Images (TI) being upper contour, lower contour and loop images.

After the feature vectors are generated by employing each technique, the features are then applied to the ANN and SVM for training and testing for the recognition/identification process. For both ANNs and SVMs settings and structures, there were 788 outputs for the 788 first and last names for the bilingual dataset and 394 outputs for the 394 first and last names for Thai and English datasets. The duplicated name components from different writers, for example "สุวรรณวิวัฒน์" from "เหมพรรณ สุวรรณวิวัฒน์" and "สุวรรณวิวัฒน์" from "ธนวัฒน์ สุวรรณวิวัฒน์" were classified into 2 different outputs. However, in the recognition phase, "สุวรรณวิวัฒน์" can be recognised as either "สุวรรณวิวัฒน์" of เหมพรรณ's output or of ธนวัฒน์'s output. The BSIS will identify to whom the name component belongs to. This is applied to both languages. Classification into 2 different outputs was carried out because, in the future, it is believed that this will be useful in developing a BSIS that can identify and verify students from their name components.

Eighty percent of the dataset was used for training and the other 20% were used for testing. That is, 6,304 samples were used for training and 1,576 samples were used for testing. All 7,880 samples were used in four-fold cross validation SVMs. For ANNs, the number of hidden units investigated was experimentally set initially at 100, incrementing upwards by 1 up to 650 hidden units. The number of iterations set for training increased from 1,000 up to 10,000, incrementing by 1,000. All ANNs were trained with MATLAB/2011a default parameters for resilient backpropagation. For the SVM experimental settings, four-fold cross validation was performed to achieve statistically meaningful results. Three kernel types, radial basis function, linear, and precomputed kernel, were used, and the C parameter of the SVM was set to be 35.

From Figure 4.18, it can be seen that the best recognition rate when employing ANNs as the classifier for the Thai dataset was 95.30%, the rate was attained when employing the MDF extraction technique, while the lowest recognition rate of 77.28% was obtained when employing WRLMDF technique. When employing most of the feature extraction techniques on TI, except for WRLMDF, the recognition rates were lowered. The WRLMDF recognition rate was improved by 2.54% (from 77.28% to 79.82%). The best recognition rate, 91.50%, obtained occurred when utilising TI, which employed MDF as the feature extraction technique.

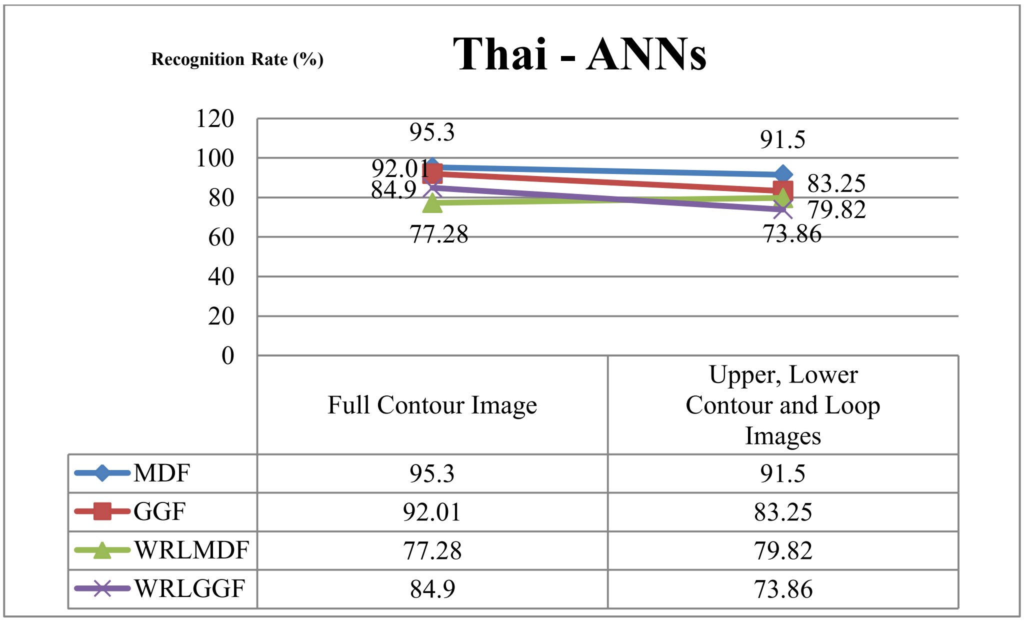


Figure 4.18: Recognition rates for each feature extraction technique employing the ANNs as the classifier for Thai name components

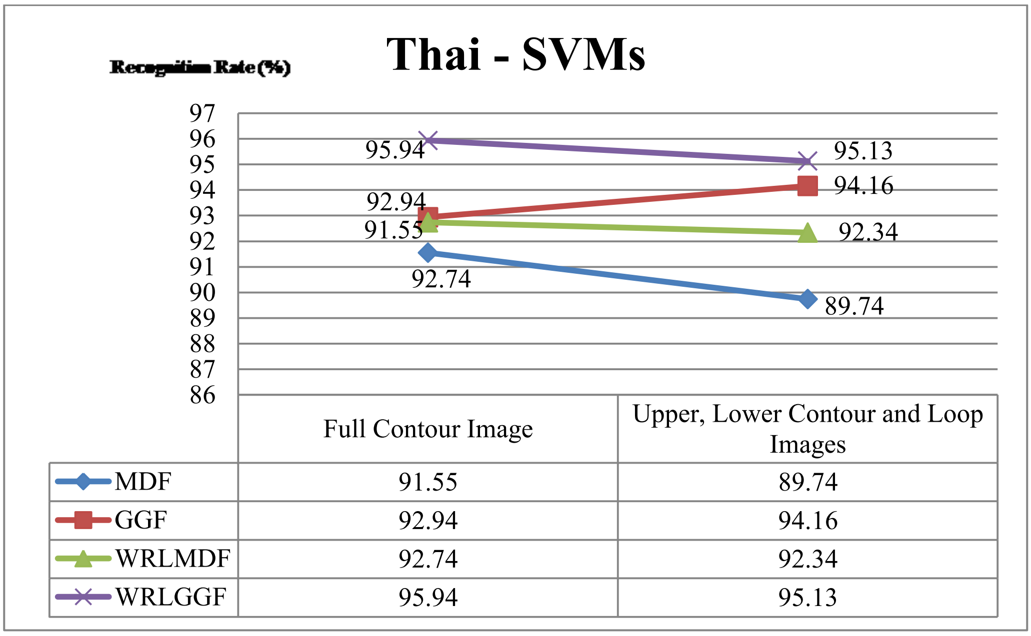


Figure 4.19: Recognition rates for each feature extraction technique employing the SVMs as the classifier for Thai name components

For Thai dataset, when employing SVMs as the classifier, it can be seen in Figure 4.19 that the best recognition rate attained was 95.94%, attained when WRLGGF was employed. This rate was 4.39% better when compared to the best recognition rate of the MDF employing ANNs as the classifier (95.30% vs. 95.94%, respectively). The lowest recognition rate obtained when employing the SVMs as the classifier was 89.74% which was obtained when the MDF was utilised on TI. Similar to the results from those employing ANNs as classifiers, when employing most of the feature extraction techniques on TI, except for GGF, the recognition rates were lowered. The GGF was improved by 1.22% (from 92.94% to 94.16%). The best recognition rate obtained when utilising TI was 95.13%, which was achieved by employing WRLGGF as the feature extraction technique.

From Figure 4.20, it can be seen that the best recognition when employing ANNs as the classifier rate for the English dataset was 96.19%, attained when employing the MDF extraction technique, while the lowest recognition rate of 85.41% was obtained when employing the WRLMDF technique. When employing most of the feature extraction techniques on TI, except for WRLMDF, the recognition rates were lowered. The WRLMDF recognition rate was improved by 1.65% (from 85.41% to 87.06%). The best recognition rate obtained when utilising TI was 92.51%, which achieved employing MDF as the feature extraction technique.

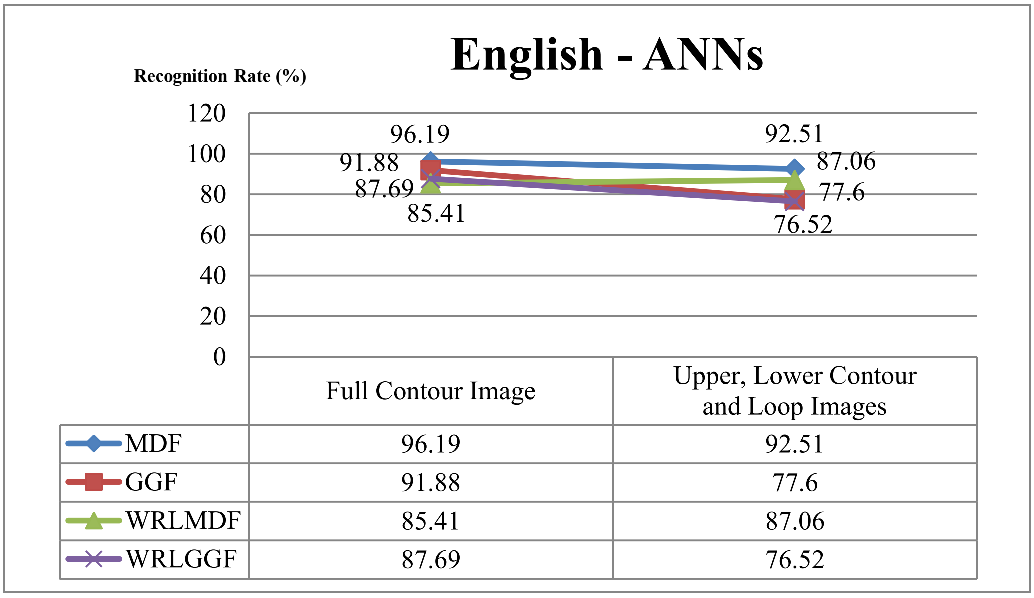


Figure 4.20: Recognition rates for each feature extraction technique employing the ANNs as the classifier for English name components

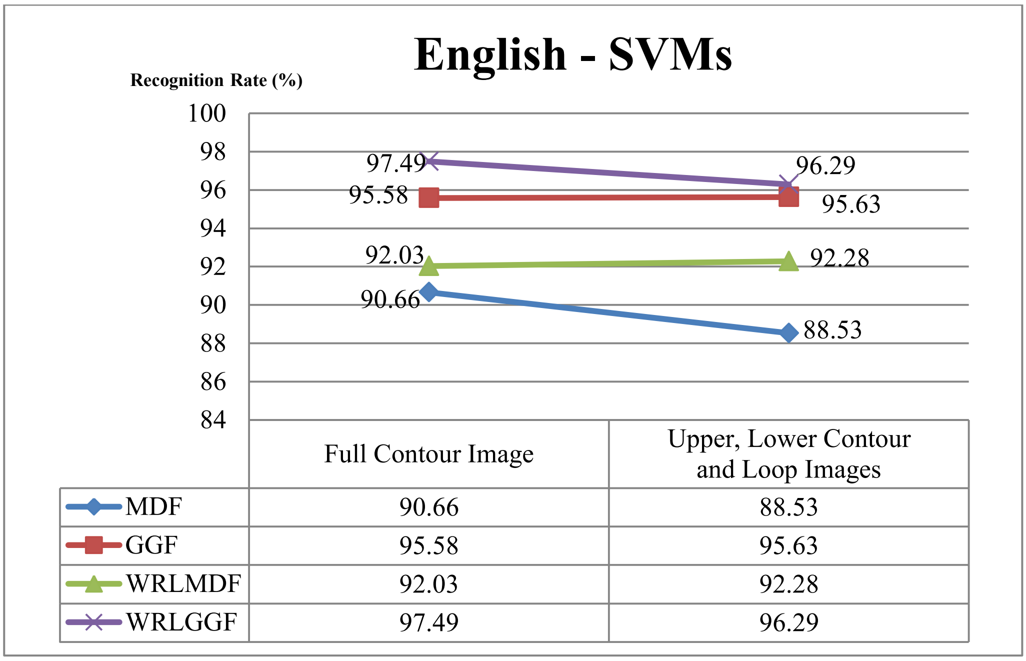
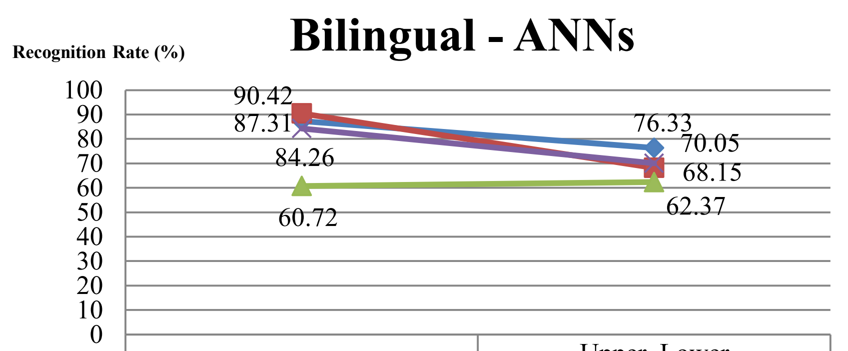


Figure 4.21: Recognition rates for each feature extraction technique employing the SVMs as the classifier for English name components

When employing SVMs as the classifier, it can be seen in Figure 4.21 that the best recognition rate attained was 97.49%, achieved when WRLGGF was employed on the full contour images. This rate was 1.30% better when compared to the best recognition rate of the MDF employing ANNs as the classifier (96.19% vs. 97.49%, respectively). The lowest recognition rate obtained when the SVMs were employed as the classifier was 90.66%, achieved when MDF was utilised. When employing SVMs as classifiers, recognition rates obtained from employing GGF and

WRLMDF on TI were higher by 0.05% and 0.25% respectively, while the recognition rates from employing MDF and WRLGGF on TI were lower by 2.13% and 1.5% respectively. The best recognition rate obtained when utilising TI was 96.29% which was from employing WRLGGF as the feature extraction technique.

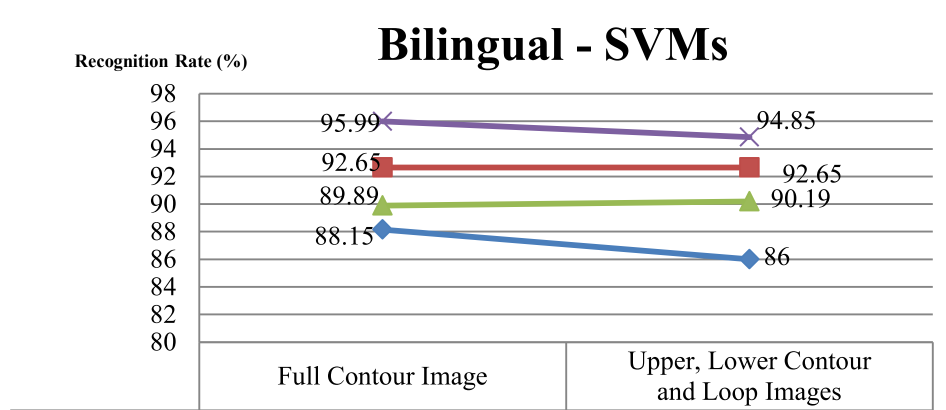


	Full Contour Image	Upper, Lower Contour and Loop Images
MDF	87.31	76.33
GGF	90.42	68.15
WRLMDF	60.72	62.37
WRLGGF	84.26	70.05

Figure 4.22: Recognition rates for each feature extraction technique employing the ANNs as the classifier for bilingual name components

From Figure 4.22, it can be seen that the best recognition rate when employing ANNs as the classifier for the bilingual dataset was 90.42%, attained when employing the GGF extraction technique, while the lowest recognition rate of 60.72% was obtained when employing the WRLMDF technique. When employing most of the feature extraction techniques on TI, except for WRLMDF, the recognition rates were lowered. The WRLMDF recognition rate was improved by 1.65% (from 60.72% to 62.37%). The best recognition rate obtained when utilising TI was 76.33% using the MDF feature extraction technique; this highest rate was, however, still lower than the highest recognition rate when employing the MDF on the full boundary contour by 10.98% (76.33% comparing to 87.31%).

When employing SVMs as the classifier, it can be seen in Figure 4.23 that the best recognition rate attained was 95.99%, achieved when WRLGGF was employed on the full boundary contour. This rate was 5.57% better when compared to the best recognition rate of the GGF employing ANNs as the classifier (95.99% vs. 90.42%, respectively). The lowest recognition rate obtained when the SVMs were employed as the classifier was 86%, achieved when MDF was utilised on TI. When employing SVMs as classifiers, recognition rates obtained from employing WRLMDF on TI was lower by 0.3%, while the recognition rates from employing MDF and WRLGGF on TI were lower by 2.15% and 1.14% respectively. No change in the recognition rate was found for GGF as its recognition rate remained at 92.65%. The best recognition rate obtained when utilising TI was 94.85% which was attained from employing WRLGGF as the feature extraction technique.



	Full Contour Image	Upper, Lower Contour and Loop Images
MDF	88.15	86
GGF	92.65	92.65
WRLMDF	89.89	90.19
WRLGGF	95.99	94.85

Figure 4.23: Recognition rates for each feature extraction technique employing the SVMs as the classifier for bilingual name components

Further analysis was carried out based on the use of ANNs as the classifier, the reason being that bigger gaps between recognition rates from each feature extraction technique can be seen more clearly. It must be noted that each of the percentages in Table 4.6, Figure 4.24-4.27 were calculated from all the ANNs setting' recognition rates combined (for each feature extraction technique) and not from the highest recognition rates from each feature extraction technique individual setting. From Table 4.6, it can be seen that from over all the feature extraction techniques, English name components were more correctly recognised by 0.13%. Thai name components were 99.21% correctly recognised compared to 99.317% % of correctly recognised English name components. The most effective feature extraction techniques which gave the best result when applied to Thai name components were GGF and WRLGGF on the full boundary contour, and WRLGGF on TI whereas GGF and WRLGGF on TI were the most effective for English name components.

Table 4.6: A comparison of the rates of Thai and English name components which are both correctly recognised (CRR) and unable to be recognised (NNR) for each feature extraction technique employing different input images (full boundary vs. Three Images (TI) in conjunction with the ANNs

FET	Correctly Recognised Rate-CRR (%)		Non Recognised Rate NRR (%)		
	Thai	*English*	*Thai*	*English*	*Total*
MDF	98.54	98.06	1.46	1.94	3.40
GGF	100.00	99.76	0.00	0.24	0.24
WRLMDF	98.06	99.02	1.94	0.98	2.92
WRLGGF	100.00	99.52	0.00	0.48	0.48
MDF-TI	99.52	99.52	0.48	0.48	0.96
GGF-TI	99.76	100.00	0.24	0.00	0.24
WRLMDF-TI	97.82	98.8	2.18	1.20	3.38
WRLGGF-TI	100.00	100.00	0.00	0.00	0.00
AVERAGE	**99.21**	**99.34**	**0.79**	**0.67**	**1.45**

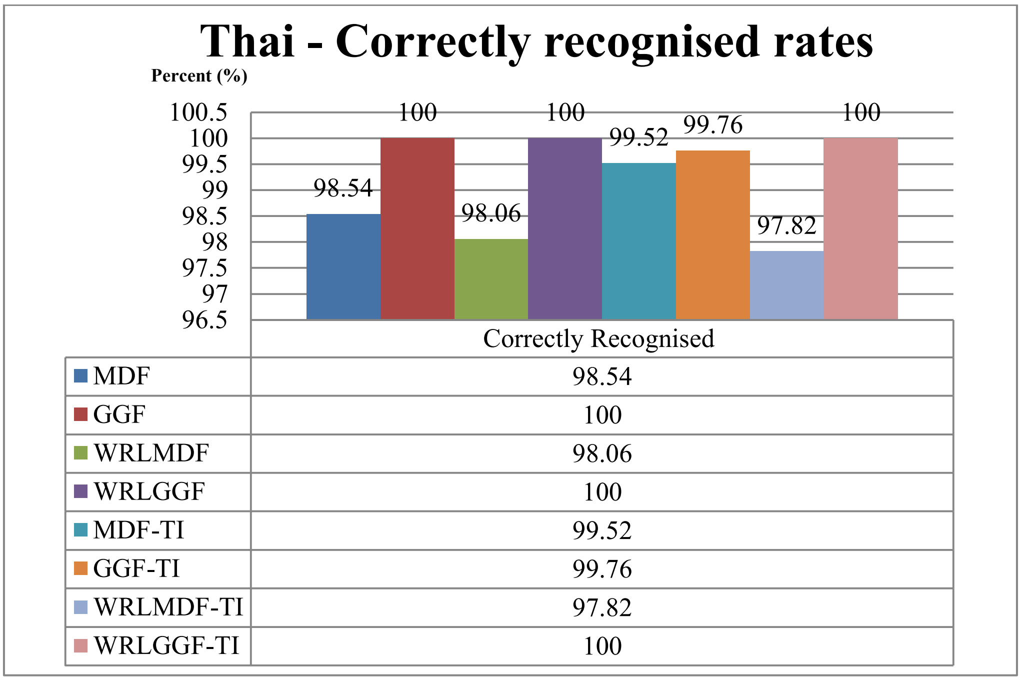
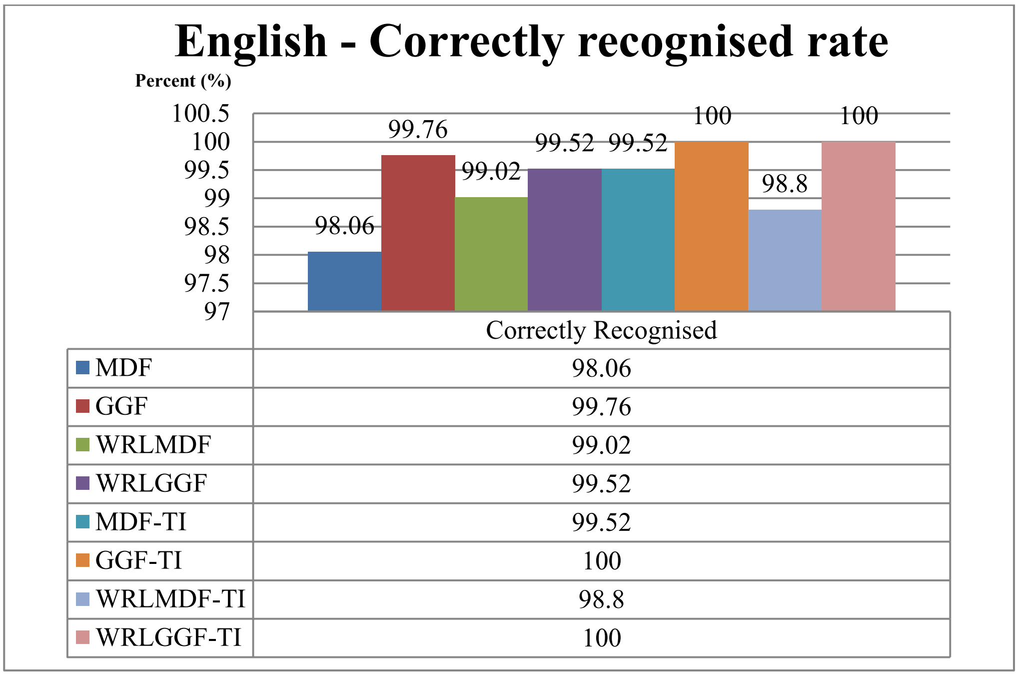


Figure 4.24: Correctly recognised rate comparisons of each feature extraction technique for Thai name components

It can be seen from Table 4.6 that 1.45% of name components comprising both Thai and English, were unable to be recognised (NNR). The percentage of Thai name components that could not be recognised is 0.79% compared to English name components of 0.67%. None of the classifiers could correctly recognise the English name component "Chawpangmon"; when looked at in detail, it was found that the writer wrote each name component quite differently, and that loops and water reservoirs were not consistent. Interrogation of input images found that the top 100 of misrecognised name components (both Thai and English) contain missing holes and some of the reservoirs could not be seen or were incomplete. As a result, lower recognition rates were obtained.



	Correctly Recognised
MDF	98.06
GGF	99.76
WRLMDF	99.02
WRLGGF	99.52
MDF-TI	99.52
GGF-TI	100
WRLMDF-TI	98.8
WRLGGF-TI	100

Figure 4.25: Correctly recognised rate comparisons of each feature extraction technique for English name components

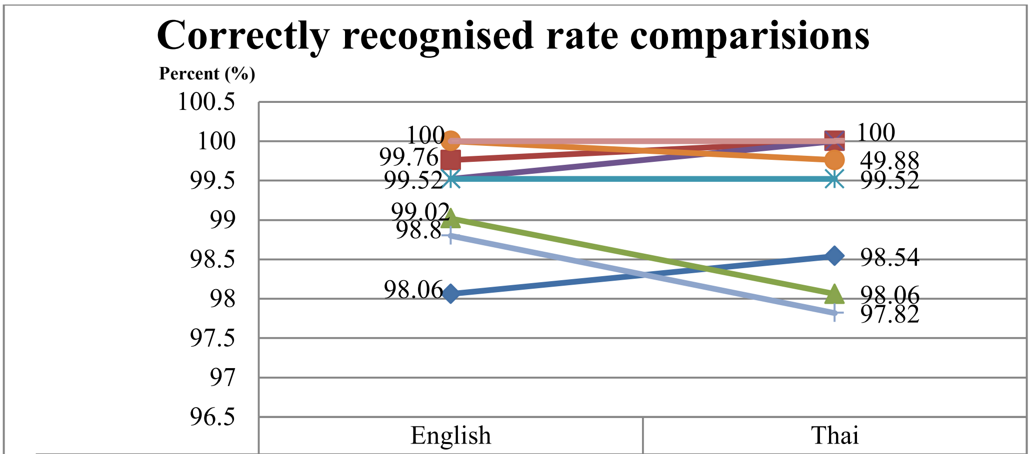
From Figure 4.24, it can be seen that for Thai language, when employing GGF or WRLGGF on full boundary images or WRLGGF on TI with all ANNs settings combined, the best ‘correctly recognised’ rate of 100% was achieved, while WRLMDF on TI yielded the lowest ‘correctly recognised’ rate of 97.82%.

Similar results to Thai language correctly recognised rates (Figure 4.24) were found for English ‘correctly recognised’ rates. As can be seen in Figure 4.25, the MDF gave the lowest ‘correctly recognised’ rate (98.06% for English and 98.54% for Thai language). GGF and WRLGGF feature extraction techniques, which were employed on TI, achieved the full score (100%) correctly recognised rate.

Figure 4.26 illustrates the ‘correctly recognised’ rate comparisons between English and Thai of each feature extraction technique, and Figure 4.27 illustrates the ‘zero-recognised’ rate comparisons between English and Thai of each feature extraction technique. From Figure 4.26, it can be seen that the feature extractions that yielded better ‘correctly recognised’ rates for the Thai name component dataset rather than the English one were MDF, GGF, and WRLGGF. The feature extraction techniques which achieved better ‘correctly recognised’ rates for English name component dataset as compared to the Thai were WRLMDF on full boundary, GGF, and WRLMDF on TI. For MDF and WRLGGF on TI, the correctly recognised rates obtained were the same for both of the languages. For English components, employing GGF and WRLGGF on

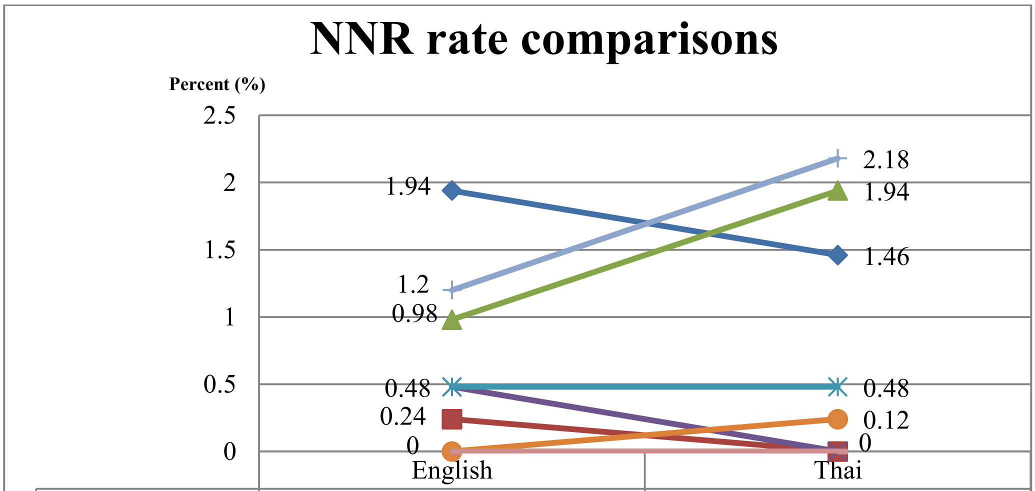
TI achieved the 'unable to recognise' rate of 0%. GGF and WRLGGF on full boundary, and WRLGGF on TI attained 0% 'unable to recognise' rate for Thai name components.

As mentioned earlier, the rates described here were obtained from every ANNs setting combined, and not from the highest recognition rates obtained from individual settings, therefore the correctly recognised rates were different from those described earlier in Figure 4.18 - 4.23, and does not reflect the highest recognition rates obtained in any way.



	English	Thai
MDF	98.06	98.54
GGF	99.76	100
WRLMDF	99.02	98.06
WRLGGF	99.52	100
MDF-TI	99.52	99.52
GGF-TI	100	99.76
WRLMDF-TI	98.8	97.82
WRLGGF-TI	100	100

Figure 4.26: Correctly recognised rate comparison between English and Thai for each feature extraction technique



	English	Thai
MDF	1.94	1.46
GGF	0.24	0
WRLMDF	0.98	1.94
WRLGGF	0.48	0
MDF-TI	0.48	0.48
GGF-TI	0	0.24
WRLMDF-TI	1.2	2.18
WRLGGF-TI	0	0

Figure 4.27: Non recognised rate comparison between English and Thai for each feature extraction technique

4.4 Experimental Results of the Proposed Off-line Short Answer Automatic Assessment System (SAAS)

The SAAS is the main sub-system of an Off-line Short answer Assessment with Student Identification System (OFSASIS), which helps to mark short answer questions with the objective of gaining greater assessment yields with minimum constraints. The proposed system employed four original feature extraction techniques namely: 1) a proposed Modified Water Reservoir, Loop and Gaussian Grid Feature (MWRLGGF); 2) Water Reservoir, Loop and Gaussian Grid Feature (WRLGGF); 3) Modified Direction Feature (MDF); and 4) the Gaussian Grid Feature extraction technique (GGF). The system also employed three proposed enhanced hybrid feature extraction techniques, namely: 1) Enhanced Water Reservoir, Loop and Gaussian Grid Feature (EWRLGGF); 2) Enhanced Gaussian Grid Feature (EGGF); 3) Enhanced Modified Direction Feature (EMDF). Consequently, the system proposed the use of seven feature extraction techniques in total. In this proposed SAAS, Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs) were employed as classifiers to compare the recognition rates. There is no publicly available examination answer dataset; as a result, a data collection process was performed to create a custom dataset. The dataset used in the present research comprises 1,248 words from 52 writers. The examination papers included 52 hand-printed (HP), and 52 cursive handwritten (CH) examination papers. In total, 1,248 (52 students × 2 handwriting types × 12 words) samples were obtained.

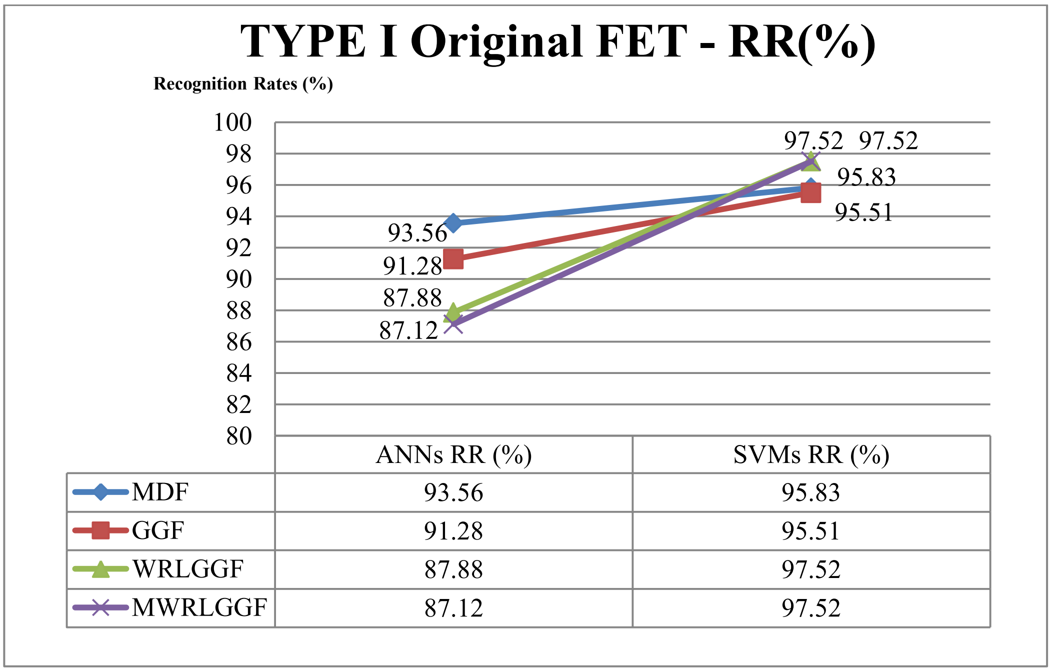
As explained in Chapter 3.1.1.A, there were 2 types of datasets, namely Type I and Type II. Type I were used for feature extraction technique recognition rate comparisons, and the datasets Type II were used for OFLAS evaluation; therefore the assessment accuracy and efficiency rates were only applied when Type II dataset was employed. All dataset samples were scanned with 300 dpi resolution and stored in grey-level format. Line and word segmentations were performed. Skew normalisation, boundary, loops, as well as upper and lower contour extractions, were performed (see Chapter 3.1.1.B for preprocessing details).

For both ANN and SVM experimental settings and structures, there were 12 outputs for the 12 answers. All types of writing of each word belong to the same output. For example "Information" and "information" were classified as the same output. For ANNs, the numbers of hidden units investigated during training were experimentally set from 20 to 120 hidden units. The number of iterations set for training increased from 500 to 10,000. For the SVM settings, a four-fold cross validation was performed to get consistent and meaningful results. The multi-class classification with a radial basis function was used and the C parameter of the SVM was set to be 35.

The results of the SAAS experiments were determined by the classification criteria. There were two minimum threshold values used as classification criteria. For threshold value ≥ 0.5 criterion, if the output's threshold was ≥ 0.5, then it was recognised as the output and eligible for the marking process, otherwise it was manually marked. For the threshold value ≥ 0.0 criterion, all the highest threshold outputs were recognised as the outputs as long as their threshold values were more than or equal to 0.0. No manual marking process took place as no minimum threshold applied (see Chapter 3.1.1.G).

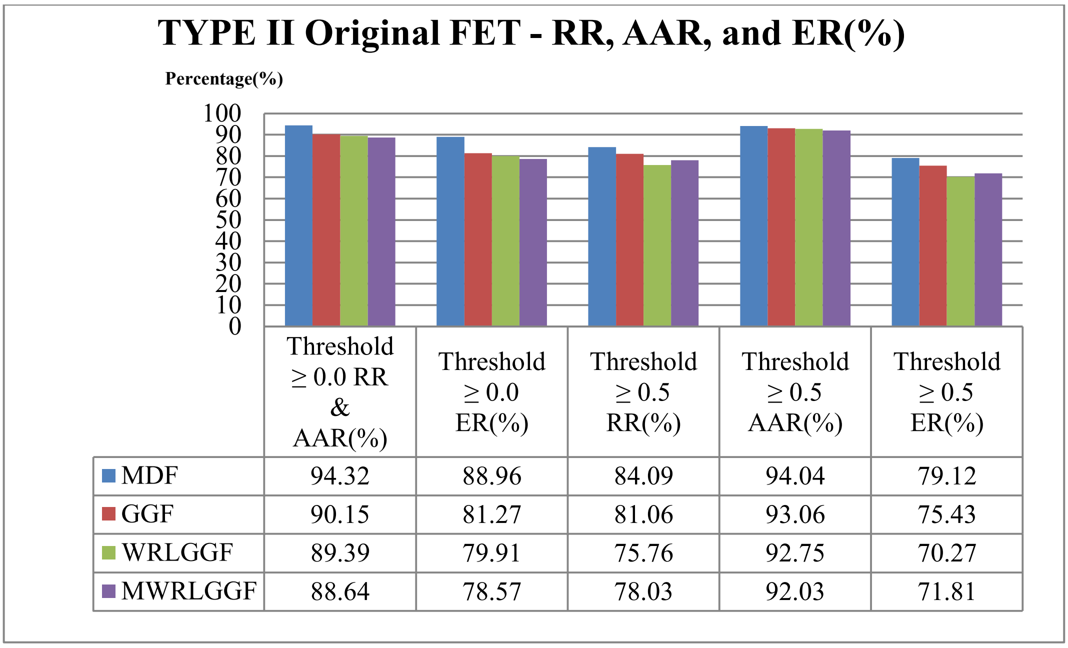
Recognition rate (RR), Assessment Accuracy Rate (AAR), and Efficiency Rate (ER) were used to evaluate the SAAS. Details of each rate can be found in Chapter 3.1.1.H. The results of each data type (I, II) and each feature extraction techniques are described as follows:

Dataset Type I were only used for recognition rate comparisons as both training and testing datasets contained only correctly spelt correct samples. It can be seen from Figure 4.28, that for this Type I dataset, the highest recognition rate was 93.56% when applying ANNs using MDF, and 97.52% when applying WRLGGF or MWRLGGF with SVMs. The MWRLGGF gave the lowest recognition rate of 87.12% when employing ANNs as the classifier; however, when SVMs was utilised, the result from employing MWRLGGF on the dataset attained the highest recognition rate of 97.52%, which is the same rate achieved when employing WRLGGF.



	ANNs RR (%)	SVMs RR (%)
MDF	93.56	95.83
GGF	91.28	95.51
WRLGGF	87.88	97.52
MWRLGGF	87.12	97.52

Figure 4.28: Recognition rates of the original feature extraction techniques employed on dataset TYPE I employing the ANNs and SVMs as classifiers



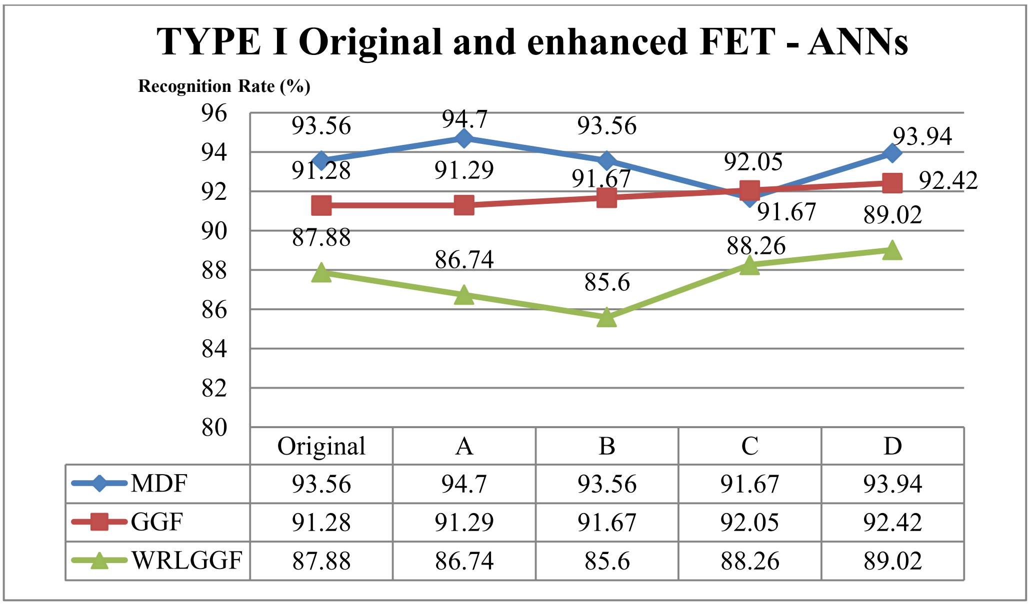
	Threshold ≥ 0.0 RR & AAR(%)	Threshold ≥ 0.0 ER(%)	Threshold ≥ 0.5 RR(%)	Threshold ≥ 0.5 AAR(%)	Threshold ≥ 0.5 ER(%)
MDF	94.32	88.96	84.09	94.04	79.12
GGF	90.15	81.27	81.06	93.06	75.43
WRLGGF	89.39	79.91	75.76	92.75	70.27
MWRLGGF	88.64	78.57	78.03	92.03	71.81

Figure 4.29: Recognition Rates (RR), Assessment Accuracy Rates (AAR), and Efficiency Rates (ER) of the original feature extraction techniques employed on dataset TYPE I employing ANNs as the classifier

It was found that the proposed MWRLGGF was comparable to the original WRLGGF. In the case where the output's threshold value was ≥ 0.5, the MWRLGGF yielded slightly better results (a 78.03% recognition rate with a 92.03% assessment accuracy rate and a 71.81% efficiency rate of MWRLGGF vs. a 75.76% recognition rate with a 92.03% assessment accuracy rate and a 71.81% efficiency rate of the original WRLGGF). However, in all cases where the threshold was ≥ 0.0, it was found that the proposed technique marginally lowered the recognition, assessment accuracy and efficiency rates, even though the loop feature vector had been modified (refer to Chapter 3.1.1.C.4). It could be assumed that by reducing the amount of extra loops which were found in each window, the meaningful and useful loops which exist in the image might be eliminated. However, since there was no constraint, such as threshold, the assessment accuracy rates were lowered.

The results of proposed enhanced feature extraction techniques namely EWRGGF, EMDF, and EGGF, together with the original WRLGGF, MDF, and GGF are displayed in Figure 4.30 – 4.35 and Table 4.7 below. It is to be noted that the feature extraction techniques were applied separately and were not combined in any way.

The results when employing original and enhanced feature extraction techniques on Type I dataset, which was used for training the ANNs and SVMs for feature extraction techniques recognition rates comparison, can be seen in Figure 4.30 – 4.31. From Figure 4.30, it was found that most of the enhanced feature extraction techniques outperformed the original feature extraction techniques. For the MDF group, EMDF_A and EMDF_D yielded better recognition rate than the original MDF by 1.14% and 0.38%, respectively (94.70% and 93.94 vs. 93.56%). With the GGF group, it was found that all of the enhanced GGF (EGGF_A, EGGF_B, EGGF_C, and EGGF_D) outperformed the original GGF by 0.01%, 0.39%, 0.77%, and 1.14%, respectively (91.29%, 91.67%, 92.05%, and 92.42% vs. 91.28%). For the WRLGGF group, EWRLGGF_C and EWRLGGF_D outperformed the original WRLGGF by 0.38% and 1.14%, respectively (88.26% and 89.02% vs. 87.88%). For dataset Type I, the best recognition rate of 94.70% was obtained when the newly proposed EMDF_A was applied to ANNs.



	Original	A	B	C	D
MDF	93.56	94.7	93.56	91.67	93.94
GGF	91.28	91.29	91.67	92.05	92.42
WRLGGF	87.88	86.74	85.6	88.26	89.02

Figure 4.30: TYPE I dataset Recognition Rate (RR), Assessment Accuracy Rate (AAR), and Efficiency Rate (ER) comparisons between each original and enhanced Feature Extraction Technique (FET) employing ANNs as the classifiers

From Figure 4.31, it was found that when employing SVMs as classifiers, all of the enhanced feature extraction techniques outperformed the original feature extraction techniques, except for EGGF_A, and WRLGGF_A where their recognition rates were equal to their original counterparts (95.51% for both GGF and EGGF_A, and 97.52% for both WRLGGF and EWRLGGF_A).

For the MDF group, EMDF_A, EMDF_B, EMDF_C, and EMDF_D yielded better recognition rates than the original MDF by 0.32%, 0.32%, 0.72% and 0.93%, respectively (96.15%, 96.15%, 96.55%, and 96.79 vs. 95.83%). With the GGF group, it was found that EGGF_B, EGGF_C, and EGGF_D) outperformed the original GGF by 0.08%, 0.08%, and 0.32%, respectively (95.59%, 95.59%, and 95.83% vs. 95.51%). For the WRLGGF group, EWRLGGF_B, EWRLGGF_C, and EWRLGGF_D outperformed the original WRLGGF by 0.08%, 0.16%, and 0.16%, respectively (97.6%, 97.68% and 97.68% vs. 97.52%). For dataset Type I, the best recognition rate of 97.68% was attained when the newly proposed EWRLGGF_C and EWRLGGF_D were applied to SVMs (compared to 97.52% from the original WRLGGF).

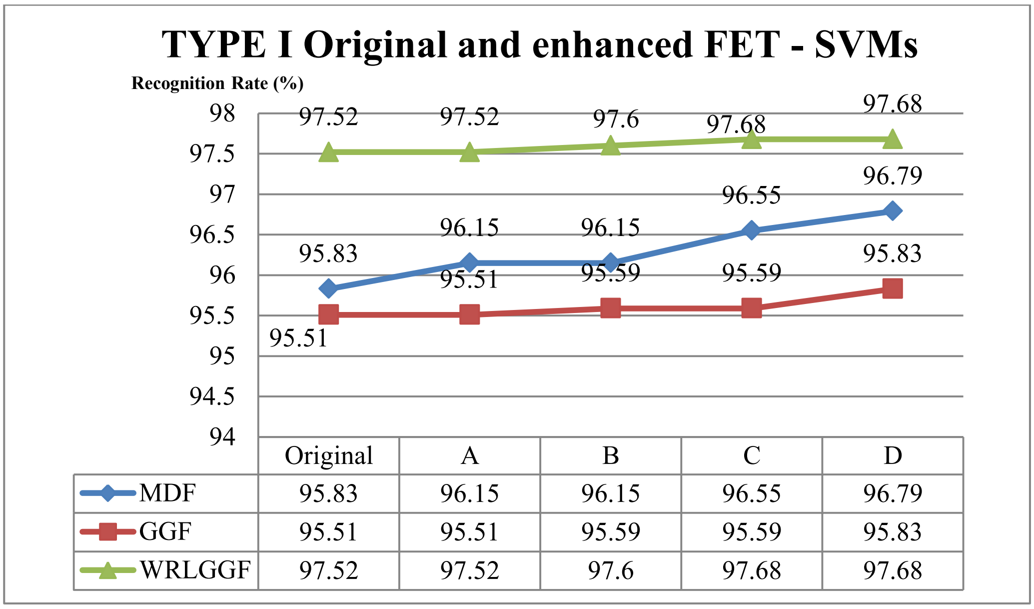


Figure 4.31: TYPE I dataset Recognition Rate (RR), Assessment Accuracy Rate (AAR), and Efficiency Rate (ER) comparisons between each original and enhanced Feature Extraction Technique (FET) employing the SVMs as classifiers

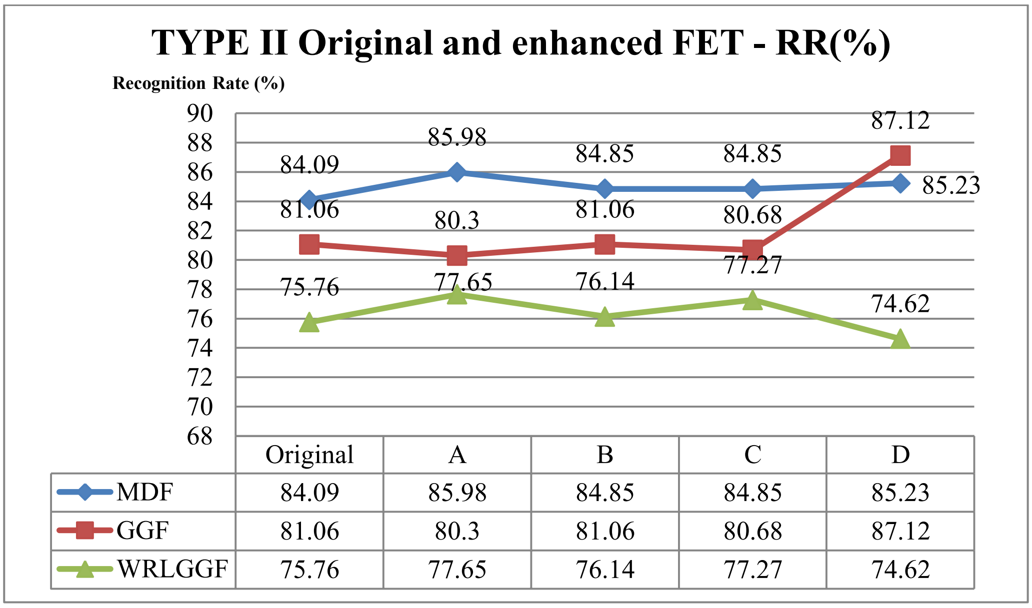


Figure 4.32: TYPE II dataset Assessment Accuracy Rate (AAR) comparisons between each original and enhanced feature extraction technique

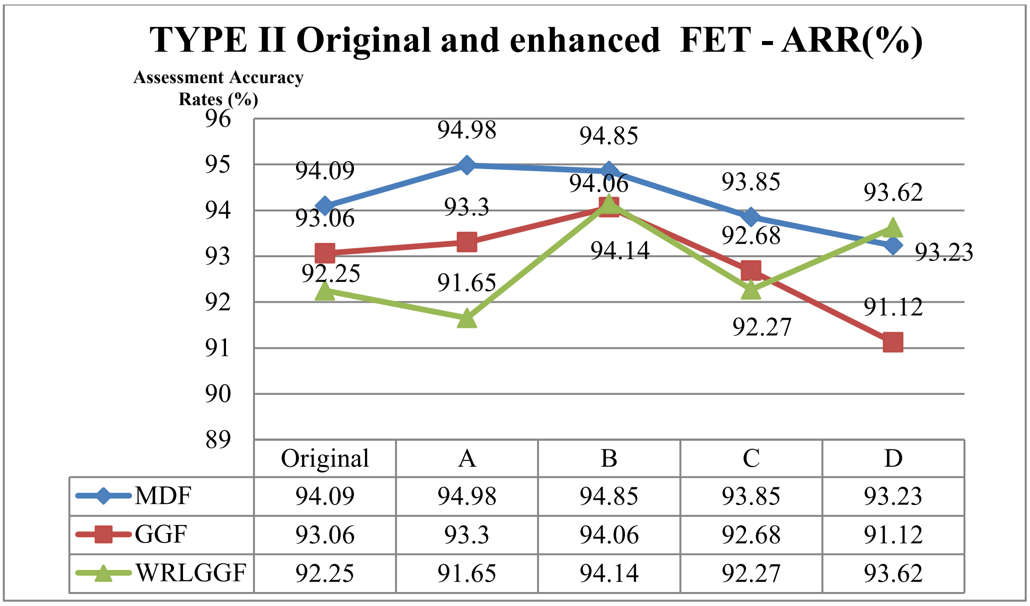
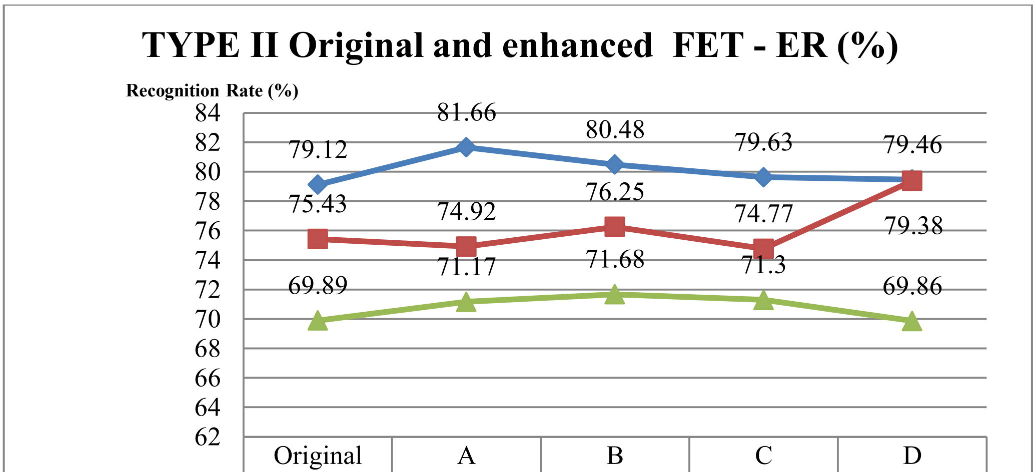


Figure 4.33: TYPE II dataset Assessment Accuracy Rate (AAR) comparisons between each original and enhanced feature extraction technique

Details of dataset Type II recognition rates, assessment accuracy rates and efficiency rates achieved from each original and enhanced feature extraction techniques can be found in Figure 4.32 -4.34. From Figure 4.32, it can be seen that for the MDF group, all of the enhanced MDF outperform the original MDF by 1.89%, 0.76%, 0.76%, and 1.14% (85.98%, 84.85%, 84.85%, and 85.53% vs. 84.09% respectively). With the GGF group, it was found only EGGF_D could outperform its original GGF; however, it had outperformed its original GGF counterpart by 6.06% (87.12% vs. 81.06%). For the WRLGGF group, except for EWRLGGF_D, all other enhanced WRLGGF (EWRLGGF_A, EWRLGGF_B, and EWRLGGF_C) were able to outperform the original WRLGGF by 1.89%, 0.38%, and 1.51%, respectively (77.65%, 76.14%, 77.27% and 75.76%). For dataset Type II, the best recognition rate of 87.12% was obtained from employing EGGF_D to the SAAS.

After observing assessment accuracy rates when feature extraction techniques were employed on dataset Type II in Figure 4.33, it was found that the highest assessment accuracy rate of 94.98% was achieved from employing EMDF_A on the dataset, while the lowest assessment accuracy rate of 91.12% was obtained from utilising EGGF_D. The enhanced MDF namely, EMDF_A and EMDF_B outperformed the original MDF by 0.89% and 0.76%, respectively (94.98% and 94.85% vs. 94.09%). The EGGF_A and EGGF_B outperformed its original counterpart by 0.24% and 1%, respectively (93.30 and 94.06% vs. 93.06%). With WRLGGF group, EWRLGGF_B, EWRLGGF_C, and EWRLGGF_D outperformed the WRLGGF by 1.89%, 0.02%, and 1.37% (94.14%, 92.27%, and 93.62% vs. 92.25% respectively).



	Original	A	B	C	D
MDF	79.12	81.66	80.48	79.63	79.46
GGF	75.43	74.92	76.25	74.77	79.38
WRLGGF	69.89	71.17	71.68	71.3	69.86

Figure 4.34: TYPE II dataset Efficiency Rate (ER) comparisons between each original and enhanced feature extraction technique

The efficiency rates of each of the feature extraction techniques for dataset Type II can be seen in Figure 4.34. As can be seen from this figure, the best efficiency rate of 81.66% was obtained from employing EMDF_A. It was observed that all enhanced MDF (EMDF_A, EMDF _B, EMDF _C, and EMDF_D) outperformed its original counterpart; the efficiency rates achieved were 81.66%, 80.48%, 79.63%, and 79.46% compared to 79.12% of the original MDF, respectively. The EGGF_B and EGGF_D outperformed the original GGF by 0.82% and 3.95%, respectively (76.25%, 79.38% vs. 75.43%). For the WRLGGF group, except for EWRLGGF_D, all other enhanced WRLGGF namely, EWRLGGF_A, EWRLGGF_B, and

EWRLGGF_C could outperform the original WRLGGF by 1.28%, 1.79%, and 1.41% respectively (71.17%, 71.68%, and 71.3 vs. 69.89% of the WRLGGF).

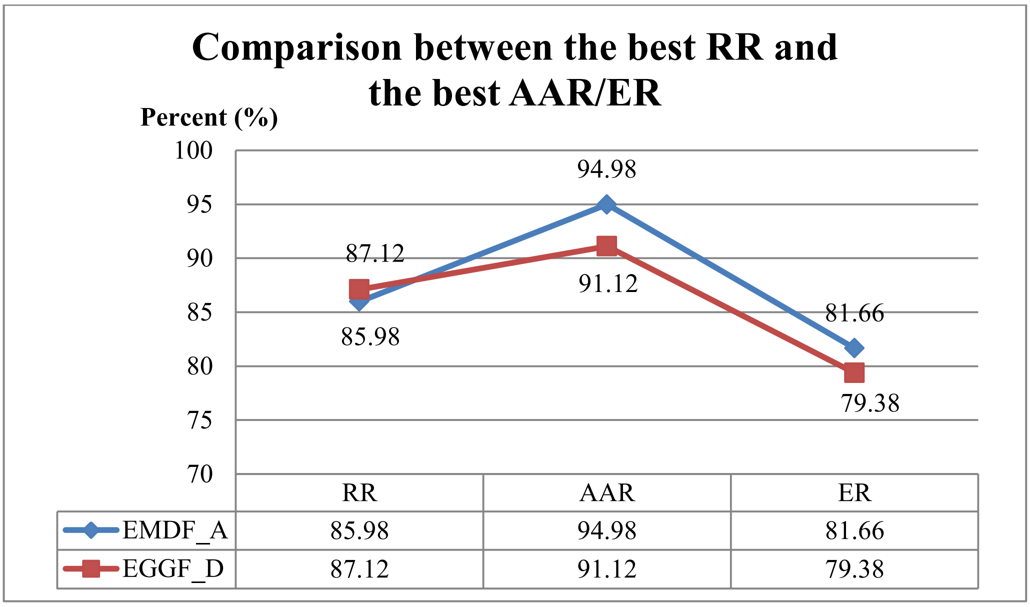


Figure 4.35: The best Recognition Rate (RR) and the best Assessment Accuracy Rate (AAR)/Efficiency Rate (ER) comparisons employing TYPE II dataset

Table 4.7: Recognition rate (RR), Assessment Accuracy Rate (AAR), and Efficiency Rate (ER) comparisons between each feature extraction technique (FET) as applied to the SAAS

FET	Dataset Type I		Dataset Type II		
	ANNs RR (%)	*SVMs RR (%)*	*RR (%)*	*AAR (%)*	*ER (%)*
MDF	93.56%	95.83%	84.09%	94.09%	79.12%
EMDF_A	**94.70%**	96.15%	85.98%	**94.98%**	**81.66%**
EMDF_B	93.56%	96.15%	84.85%	94.85%	80.48%
EMDF_C	91.67%	96.55%	84.85%	93.85%	79.63%
EMDF_D	93.94%	96.79%	85.23%	93.23%	79.46%
GGF	91.28%	95.51%	81.06%	93.06%	75.43%
EGGF_A	91.29%	95.51%	80.30%	93.30%	74.92%
EGGF_B	91.67%	95.59%	81.06%	94.06%	76.25%
EGGF_C	92.05%	95.59%	80.68%	92.68%	74.77%
EGGF_D	92.42%	95.83%	**87.12%**	91.12%	79.38%
WRLGGF	87.88%	97.52%	75.76%	92.25%	69.89%
EWRLGGF_A	86.74%	97.52%	77.65%	91.65%	71.17%
EWRLGGF_B	85.60%	97.60%	76.14%	94.14%	71.68%
EWRLGGF_C	88.26%	**97.68%**	77.27%	92.27%	71.30%
EWRLGGF_D	89.02%	**97.68%**	74.62%	93.62%	69.86%
MWRLGGF	87.12%	97.52%	78.03%	92.03%	71.81%

From Figure 4.35, it can be seen that even though the best recognition rate of 87.12% was obtained when EGGF_D was applied to the dataset Type II, its assessment accuracy and efficiency rates could not outperform the assessment accuracy and efficiency rates attained from employing EMDF_A. The recognition rate obtained from the EMDF_A was 85.98%, which was lower than the recognition rate achieved from employing EGGF_D by 1.14%. However, the EMDF_A assessment accuracy rate was 94.98%, which was 3.86% higher than the rate of 91.12% that was obtained from the EGGF_D. The EMDF_A yielded 81.66% efficiency rate which is also higher than the efficiency rate obtained from employing EGGF_D by 2.28%. The conclusion results of the Type II dataset can be seen in following Table 4.7.

Chapter 5

ANALYSIS AND COMPARISION OF RESULTS

From Chapter 4, it can be seen that a substantial number of experiments have been performed in this research. The purpose of these experiments was to investigate 1) the performance of state-of-art feature extraction techniques, namely MDF, GGF, WR, and 2) newly proposed feature extraction techniques, namely, WRLGGF, WRLMDF, and enhanced MDF, GGF, WRLGGF, and WRLMDF for the problem of word recognition (Thai, English, and bilingual name components and short answer words).

Different datasets in terms of language and dataset size, types of input image, different feature extraction techniques, and classifiers employed have been utilised in this research. It can be considered difficult to compare the performance of the proposed systems, especially the Student Identification Systems (SIS), to other systems as to the best of the author's knowledge, there is no SIS found in the literature. The Short Answer automated Assessment System (SAAS) can be compared to some studies found in the literature. For both the SIS and SAAS, the dataset sizes used in this research study were different than those in the literature.

This chapter is organised as follows: Section 5.1 presents the analysis of English SIS results, while Section 5.2 compares the results of Thai SIS of each dataset size. Section 5.3 presents an analysis of the results and performance of BSIS. The comparison and analysis of overall SISs and the word recognition systems found in the literature are presented in Section 5.4. The SAAS results analysis and comparison can be found in Section 5.5. SAAS comparisons to automatic assessment systems found in the literature are described in Section 5.6.

5.1 Experimental Results Comparison and Analysis of the Proposed English Student Identification System (ESIS)

Comparisons of results attained when employing different dataset sizes, feature extraction techniques, types of input image, and classifiers were performed: details of the findings are described as followed. ESIS name component dataset sizes employed in this research were 1,040, 2,040, and 3,940. Feature Extraction Techniques (FET) which were employed in the experiments being: 1) MDF; enhanced MDF, namely, 2) EMDF_A; 3) EMDF_B; 4) EMDF_C; and 5) EMDF_D; 6) GGF; enhanced GGF, namely, 7) EGGF_A; 8) EGGF_B; 9) EGGF_C; 10) EGGF_D; 11) WRLGGF; enhanced WRLGGF, namely, 12) EWRLGGF_A; 13) EWRLGGF_B; 14) EWRLGGF_C; 15) EWRLGGF_D; and 16) WRLMDF. Input images types which were utilised in the ESIS were Full Boundary Contour (FBC) and Three Images (TI) being upper contour, lower contour, and loop images; the classifiers employed in the experiments were ANNs and SVMs.

From Table 5.1, it can be seen that for ESIS name component dataset size of 1,040 that the best recognition rate of 99.52% was attained when the proposed EMDF_A and _C was applied to ANNs. It was also observed that the same recognition rate (99.52%) was obtained when the original WRLGGF, together with its enhanced FET (EWRLGGF_A-D) were employed. The experiments on 1,040 name component dataset size employed only FBC images. For the dataset size of 2,040 name component samples, the best recognition rate achieved was 99.02% which was attained when MDF was employed on FBC and when it was applied to ANNs. With the 3,940 ESIS dataset size, employing the proposed WRLGGF on FBC yielded the best recognition rate of 97.49%.

Figure 5.1 and Table 5.2 illustrate a comparison between each of the datasets best recognition rates. The 1,040 name component dataset, which is the smallest of the three (1,040, 2,040, and 3,940), yielded the highest recognition rate of 99.52%. The recognition rates dropped to 99.02% and 97.50% for 2,040 and 3,940 name components respectively, or in other words, they were 0.50% and 2.03% lower than 99.52% of the 1,040 dataset size, respectively. Table 5.2 shows that both of the classifiers (ANNs and SVMs) yielded the best recognition rates, although with different datasets. It can be noted that ANNs and SVMs yielded the same best recognition rate for the 1,040 dataset size, although employing different FETs.

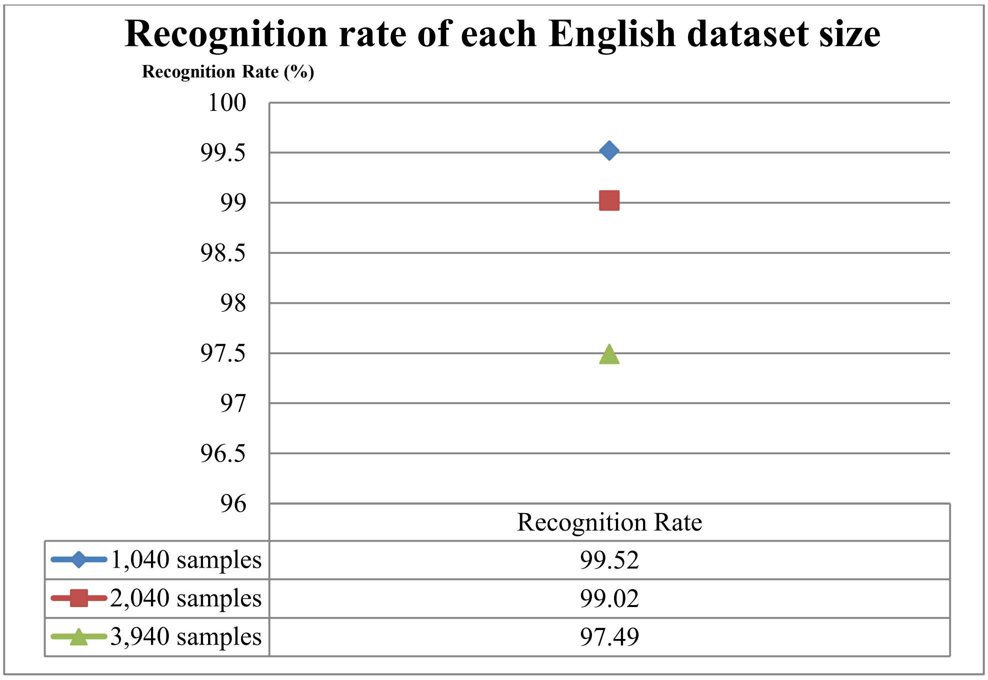


Figure 5.1: Recognition rates of each of the ESIS dataset size

Table 5.1: Each of English name component dataset size with recognition rates obtained, together with the feature extraction techniques and image type employed in conjunction with ANNs and SVMs as classifiers

Dataset Size	FET	Image Type	Recognition Rate (%)	
			ANNs	*SVMs*
1,040	MDF	FBC	99.04	97.31
1,040	EMDF_A	FBC	99.04	97.12
1,040	EMDF_B	FBC	**99.52**	97.21
1,040	EMDF_C	FBC	**99.52**	97.21
1,040	EMDF_D	FBC	99.04	97.04
1,040	GGF	FBC	98.08	98.85
1,040	EGGF_A	FBC	98.08	98.85
1,040	EGGF_B	FBC	98.08	98.85
1,040	EGGF_C	FBC	98.08	98.85
1,040	EGGF_D	FBC	98.08	98.85
1,040	WRLGGF	FBC	95.67	**99.52**
1,040	EWRLGGF_A	FBC	95.67	**99.52**
1,040	EWRLGGF_B	FBC	95.67	**99.52**
1,040	EWRLGGF_C	FBC	95.67	**99.52**
1,040	EWRLGGF_D	FBC	95.67	**99.52**
2,040	MDF	FBC	**99.02**	N/A
2,040	GGF	FBC	98.28	N/A
3,940	MDF	FBC	**96.19**	90.99
3,940	GGF	FBC	91.88	95.58
3,940	MDF	TI	92.51	88.53
3,940	GGF	TI	77.60	95.63
3,940	WRLMDF	FBC	85.41	92.03
3,940	WRLGGF	FBC	87.69	**97.49**
3,940	WRLMDF	TI	87.06	92.28
3,940	WRLGGF	TI	76.52	96.29

Table 5.2: The best recognition rate of each of the English name component dataset size with recognition rates attained, together with the feature extraction techniques, image type, and the classifier employed

Dataset Size	FET	Contour Type	Recognition Rate (%)	Classifier	
				ANNs	SVMs
1,040	EMDF_B-C, WRLGGF, EWRLGGF_A, _D	FBC	99.52	√	√
2,040	MDF	FBC	99.02	√	
3,940	WRLGGF	FBC	97.49		√

5.2 Experimental Results Comparison and Analysis of the proposed Thai Student Identification System (TSIS)

Comparisons of results achieved when employing different dataset sizes, feature extraction techniques, types of input image, and classifiers were performed, details of the findings are described as follows. ESIS name component dataset sizes employed in this research were 2,060, and 3,940. FETs employed in the experiments were: 1) MDF; 2) GGF; 3) WRLGGF; and 4) WRLMDF. Input images types which were utilised in the ESIS were 1) FBC, 2) TI, and 3) Upper contour, Lower Contour images (ULC); the classifiers employed in the experiments were ANNs and SVMs.

From Table 5.3, it can be seen that for TSIS dataset size of 2,060 name components that the best recognition rate of 99.27% was attained when MDF and GGF was employed on TI, applied to ANNs and SVMs (separately). For the name component dataset size of 3,940 samples, the best recognition rate achieved was 95.94% which was attained when the proposed WRLGGF was employed on FBC and when it was applied to ANNs.

Figure 5.2 and Table 5.4 illustrate a comparison between each of the datasets best recognition rates. The 2,060 name component dataset, which is the smallest of the two (2,060 vs. 3,940), yielded the highest recognition rate of 99.27%. The recognition rates dropped to 95.94% when 3,940 sample dataset was employed (i.e., the recognition rate had dropped by 3.33%). Table 5.2 shows that both of the classifiers (ANNs and SVMs) yielded the best recognition rates, although with different datasets. It was observed that the 2,060 name component dataset applied to ANNs and SVMs yielded the same best recognition rate, however, by employing different FETs.

It was observed that for both datasets, both of the classifiers yielded comparable recognition rates (99.27% from both ANNs and SVMs for the 2,060 sample dataset, and 95.30% and 95.94% for the 7,880 sample dataset). From the observation, it can be assumed that for the bilingual dataset, both of the classifiers are suitable to be employed in further research in the proposed system or in word recognition tasks.

Table 5.3: Each of Thai name component dataset size with recognition rates obtained, together with the feature extraction techniques and image type employed in conjunction with ANNs and SVMs as classifiers

Dataset Size	FET	Image Type	Recognition Rate (%)	
			ANNs	*SVMs*
2,060	MDF	FBC	99.03	98.35
2,060	GGF	FBC	97.33	98.40
2,060	MDF	TI	**99.27**	97.52
2,060	GGF	TI	80.58	**99.27**
2,060	MDF	ULC	97.57	95.73
3,940	MDF	FBC	**95.30**	91.55
3,940	GGF	FBC	92.01	92.94
3,940	MDF	TI	91.50	89.74
3,940	GGF	TI	83.25	94.16
3,940	WRLMDF	FBC	77.28	92.74
3,940	WRLGGF	FBC	84.90	**95.94**
3,940	WRLMDF	TI	79.82	92.34
3,940	WRLGGF	TI	73.86	95.13

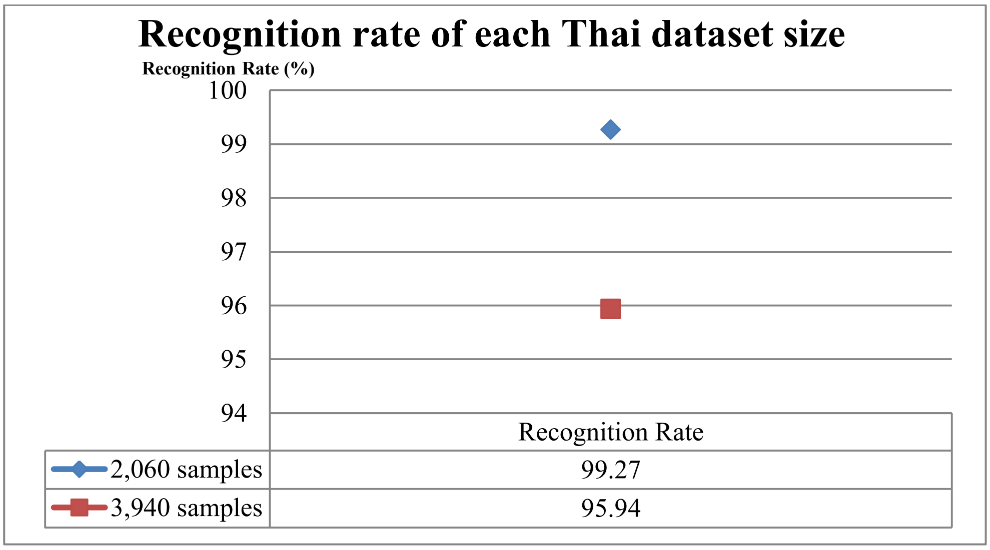


Figure 5.2: Recognition rates of each of the TSIS dataset size

Table 5.4: The best recognition rate of each of the Thai name component dataset size with recognition rates attained, together with the feature extraction techniques, image type, and the classifier employed

Dataset Size	FET	Contour Type	Recognition Rate (%)	Classifier	
				ANNs	SVMs
2,060	MDF, GGF	TI	99.27	√	√
3,940	WRLGGF	FBC	95.94		√

5.3 Experimental Results Comparison and Analysis of the Proposed Bilingual Student Identification System (BSIS)

This section presents comparisons of results achieved when employing different dataset sizes, feature extraction techniques, types of input image, and classifiers; details of the findings are described as follows. BSIS name component dataset sizes employed in this research were 2,060, and 7,880. FETs which were employed in the experiments were: 1) MDF; 2) GGF; 3) WRMDF; 4) WRGGF; 5) LMDF; 6) LGGF; 7) WRLMDF; and 8) WRLGGF. Input images types which were utilised in the BSIS were FBC and TI; the classifiers employed in the experiments were ANNs and SVMs.

Figure 5.3 and Table 5.6 illustrate a comparison between each of the datasets best recognition rates. The smallest dataset (the 2,060 name component dataset) of the two (2,060 vs. 7,880), achieved the highest recognition rate of 99.25%. The recognition rate dropped by 3.26% (from 99.25% to 99.25%) when the 3,940 sample dataset was utilised. For both datasets, the proposed WRLGGF achieved the best recognition rates when compared with the aforementioned FETs employed.

As can be seen from Table 5.6, the best recognition rate obtained by applying the feature vector obtained from employing MDF on FBC was 84.83%. This rate was 14.42% lower than the recognition rate of 99.25% obtained from employing WRLGGF on FBC, applied to the SVMs. Furthermore, this rate of 84.83% was 5.59% lower than the recognition rate of 90.42% obtained from employing the bigger dataset size of 7,880 samples (2,060 vs. 7,880). The different recognition rate of 5.57% between the two datasets (5,820 samples different) was found from employing the SVMs as the classifier on the datasets (99.25% vs. 95.99% for 2,060 and 7,880 sample datasets respectively). The reason that the low recognition rate obtained from employing ANNs on the 2,060 dataset was due to ANNs parameters employed while undertaking the experiments. Table 5.5 shows that for BSIS, applying SVMs as the classifier yielded the best recognition rates for both of the datasets.

Table 5.5: The best recognition rate of each of the bilingual name component dataset size with recognition rates attained, together with the feature extraction techniques, image type, and the classifier employed

Dataset Size	FET	Contour Type	Recognition Rate (%)	Classifier	
				ANNs	SVMs
2,060	WRLGGF	FBC	99.25		√
7,880	WRLGGF	FBC	**95.99**		√

Table 5.6: Each of bilingual name component dataset size with recognition rates obtained, together with the feature extraction techniques and image type employed in conjunction with ANNs and SVMs as classifiers

Dataset Size	FET	Image Type	Recognition Rate (%)	
			ANNs	SVMs
2,060	MDF	FBC	**84.83**	96.63
2,060	GGF	FBC	69.66	98.59
2,060	WRMDF	FBC	80.22	93.62
2,060	WRGGF	FBC	63.95	99.17
2,060	LMDF	FBC	84.71	94.59
2,060	LGGF	FBC	68.45	98.81
2,060	WRLMDF	FBC	79.13	95.40
2,060	WRLGGF	FBC	78.15	**99.25**
7,880	MDF	FBC	87.31	88.15
7,880	GGF	FBC	**90.42**	92.65
7,880	MDF	TI	76.33	86.00
7,880	GGF	TI	68.15	92.65
7,880	WRLMDF	FBC	60.72	89.89
7,880	WRLGGF	FBC	84.26	**95.99**
7,880	WRLMDF	TI	62.37	90.19
7,880	WRLGGF	TI	70.05	94.85

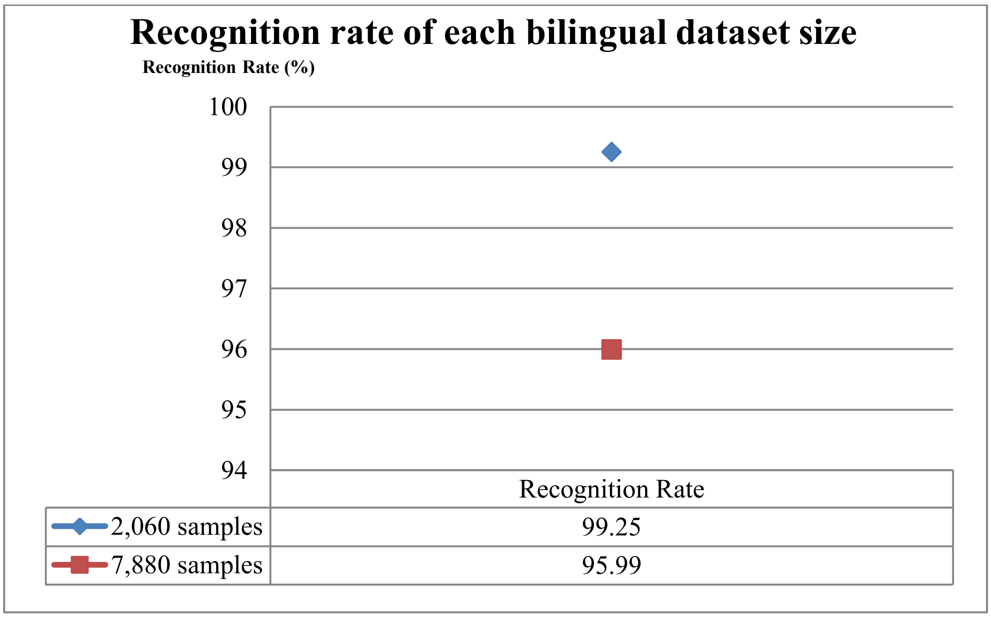


Figure 5.3: Recognition rates of each of the BSIS dataset size

5.4 SIS Results Comparison and Analysis to the Word Recognition Results Found in The Literature

To the best of the author's knowledge, there is no other work related to student identification systems available in the literature. As a consequence, the comparisons of recognition rates can only be performed with other off-line word recognition techniques found in the literature which have various database sizes (Table 5.7).

The smallest dataset of 1,040 samples employed in the proposed ESIS achieved the recognition rate of 99.52% with 100% accuracy compared to 54% valid classification rate (the system scored over 54% of all response with 99% confidence) of the automated assessment system proposed in [9]. Dataset sizes of 2,040 and 2,060 samples achieved the recognition rates of 99.02% and 99.27% compared to the recognition rate of 88.77% of the dataset size of 2,000 samples [196].

The proposed ESIS and TSIS which contained 3,940 samples in each of the datasets yielded the best recognition rates of 97.49% and 95.94% respectively. These rates were comparable to 93.50%, the recognition rate of the off-line word handwriting recogniser [197] which contained 6,410 words. The largest dataset of 7,880 samples employed in the proposed ESIS achieved the recognition rate of 99.52% with 100% accuracy compared to 72.00%, the recognition rate of bank cheque legal amount recognition system employing 5,322 samples.

As can be seen from the table and the comparisons, the feature extraction techniques proposed in this thesis are comparable to the approaches available in the literature.

Table 5.7: Results comparison between the proposed SISs and the word recognition found in the literature

System	FET	Recognition Rate (%)	Dataset Size (samples)
Off-line Whole Word Handwriting Recogniser [196]	Word Wraping	88.77	2,000
Off-line Word Handwriting Recogniser [197]	Hough Transform	93.50	3,410
Automated Assessment System [9]	HVBC	54.00	1,077
Bank Cheque Legal Amount Recognition System [109]	Stroke-and Grapheme-Based Combined Classifier	72.00	5,322
Provinces in Karnataka State Recognition System [198]	Identification of Dots, Loops, Lines, Junction points, and Endpoints	90.00	300
The Proposed English Student Identification System (ESIS)	See Section 5.1	99.52	1,040
The Proposed English Student Identification System (ESIS)	See Section 5.1	99.02	2,040
The Proposed English Student Identification System (ESIS)	See Section 5.1	97.49	3,940
The Proposed Thai Student Identification System (TSIS)	See Section 5.2	99.27	2,060
The Proposed Thai Student Identification System (TSIS)	See Section 5.2	95.94	3,940
The Proposed Bilingual Student Identification System (BSIS)	See Section 5.3	99.25	2,060
The Proposed Bilingual Student Identification System (BSIS)	See Section 5.3	95.99	7,880

5.5 Experimental Results Comparison and Analysis of the Proposed Short Answer automated Assessment System (SAAS)

This section presents experimental comparison and analysis of the proposed SAAS. As described in Chapter 3 and 4, Type I and Type II datasets were employed in SAAS experiments. While datasets Type I were used for feature extraction technique recognition rate comparisons, datasets Type II were used for OFLAS evaluation; therefore the assessment accuracy and efficiency rates were only applied when Type II dataset was employed.

For both dataset Types, comparisons of results attained when employing different handwriting styles, feature extraction techniques, types of input image, and classifiers were performed and details of the findings are described as follows. The short examination word dataset contained 1,248 samples; there were 3 types of handwritings, namely hand-printed (HP), cursive handwritten (CH), and hand-printed and cursive handwritten words combined (HPCHC). Feature Extraction Techniques (FET) which were employed in the experiments were: 1) MDF; enhanced MDF, namely, 2) EMDF_A; 3) EMDF_B; 4) EMDF_C; and 5) EMDF_D; 6) GGF; enhanced GGF, namely, 7) EGGF_A; 8) EGGF_B; 9) EGGF_C; 10) EGGF_D; 11) WRLGGF; enhanced WRLGGF, namely, 12) EWRLGGF_A; 13) EWRLGGF_B; 14) EWRLGGF_C; 15) EWRLGGF_D; 16) WRGGF; 17) LGGF; and 18) MWRLGGF. Input images types which were utilised in the SAAS were FBC, TI, and ULC, and the classifiers employed in the experiments were ANNs and SVMs.

5.5.1 Dataset Type I Result Comparison and Analysis

For dataset Type I, it can be seen from Table 5.8 and Figure 5.4 that the highest recognition rate achieved was 99.24%, achieved when applying ANNs employing feature vector obtained from MDF on HP samples. The second highest recognition rate was 97.68% which was attained by employing the proposed EWRLGGF_C and EWRLGGF_D to FBC of HPCHC samples, applied to the SVMs. The lowest best recognition rate of all handwriting types of 95.03% was obtained for CH by employing WRGGF on FBC, applied to SVMs. Figure 5.4 illustrates the gaps between each of the handwriting recognition rates.

From the recognition rates achieved from each handwriting types, it may be assumed that with all FETs, CH handwriting type could be considered hardest to recognised, and that it may have caused the recognition rate of HPCHC to lower as the HPCHC also contains CH handwriting type. To prove whether this hypothesis/assumption is the case, further investigation on larger dataset of both languages will have to be performed in future work. From Table 5.8, it can be seen that both ANNs and SVMs were both able to yield the best recognition rates, however, with different handwriting types. While ANNs yielded the best recognition rate for HP handwriting type, SVMs were able to achieve the best recognition rates for CH and HPCHC handwriting types.

Table 5.8: The best recognition rates of each handwriting type of dataset Type I

Handwriting Type	FET	Image Type	Recognition Rate (%)	Classifier	
				ANNs	SVMs
	MDF, GGF	FBC			
HP	GGF	ULC	99.24	√	
CH	WRGGF	FBC	95.03		√
HPCHC	EWRLGGF_C, _D	FBC	97.68		√

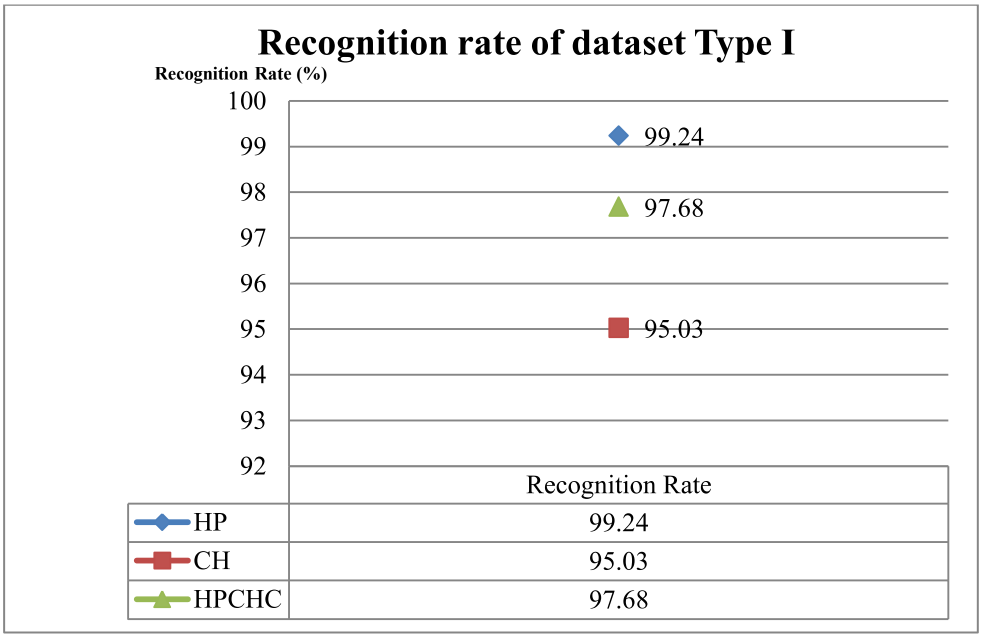


Figure 5.4: Recognition rates of each of dataset Type I

5.5.2 Dataset Type II Result Comparison and Analysis

Dataset Type II results comparison and analysis are described as follows. For dataset Type II, as described in Chapter 3 and 4, there were 2 threshold criteria employed in the experiments which were 1) threshold ≥ 0.0 and 2) threshold ≥ 0.5. All the recognition, assessment accuracy and efficiency rates were attained by employing ANNs as the classifier. Also as described in Chapter 3 and 4, the dataset Type II, which employed threshold ≥ 0.0 criterion, recognition and assessment rates were the same as no minimum threshold value was applied.

5.5.2.1 Dataset Type II Employing Threshold ≥ 0.0 Criterion

As can be seen in Table 5.9 and Figure 5.5, the highest recognition and accuracy rates amongst the three handwriting types was 97.67% which was achieved from employing GGF on TI of the HPCHC dataset, while the second highest recognition and accuracy rates of the CH dataset of 93.93% which was obtained from employing MDF on FBC. The lowest recognition and accuracy rates of the three handwriting types was 93.18% which was obtained from employing MDF to FBC of HP dataset. The difference between the second lowest and the lowest recognition and accuracy rates was, however, only 0.75% while the differences between 1) the highest and the second highest, and 2) the second highest and the lowest were 3.74% and 4.49%, respectively. From these results, it can be seen that the HPCHC handwriting type yielded the highest recognition rate. The reason was because the HPCHC dataset contained samples from both the HP and CH datasets, and therefore, more samples were available for training and testing processes. As a result, the highest recognition rate of the three kinds of writing was achieved.

The efficiency rates that were obtained are shown in Table 5.10. The highest to the lowest efficiency rates were ranked according to the recognition and accuracy rates as ranked in Table 5.9. The highest efficiency rate was 89.53% obtained from employing GGF on TI of HPCHC samples, followed by the 88.23% efficiency rate of CH samples utilising MDF on FBC; the lowest efficiency rate of the three handwriting types was 86.83% attained from employing GGF on FBC of HP samples.

When comparing the highest recognition and assessment accuracy rates in Table 5.9 to the highest efficiency rates in Table 5.10 and Figure 5.6, it can be observed that even though the recognition can be considered reasonable high (93.18-97.67%), their efficiency rates were lower than the recognition and assessment accuracy rates by 6.35% for the HP samples, 5.7% for the CH samples, and 8.14% for the HPCHC samples. In other words, the high recognition rates obtained did not reflect that the recognised words (answers) were correctly recognised, and therefore lowered the efficiency rates.

Table 5.9: The best recognition and assessment accuracy rates of each handwriting type of dataset Type II with threshold ≥ 0.0, employing ANNs as the classifier

Handwriting Type	FET	Image Type	Recognition & Assessment Accuracy Rate (%)
HP	GGF	FBC	93.18
CH	MDF	FBC	93.93
HPCHC	GGF	TI	97.67

Table 5.10: The best efficiency rates of each handwriting type of dataset Type II with threshold ≥ 0.0

Handwriting Type	FET	Image Type	Efficiency Rate (%)
HP	GGF	FBC	86.83
CH	MDF	FBC	88.23
HPCHC	GGF	TI	89.53

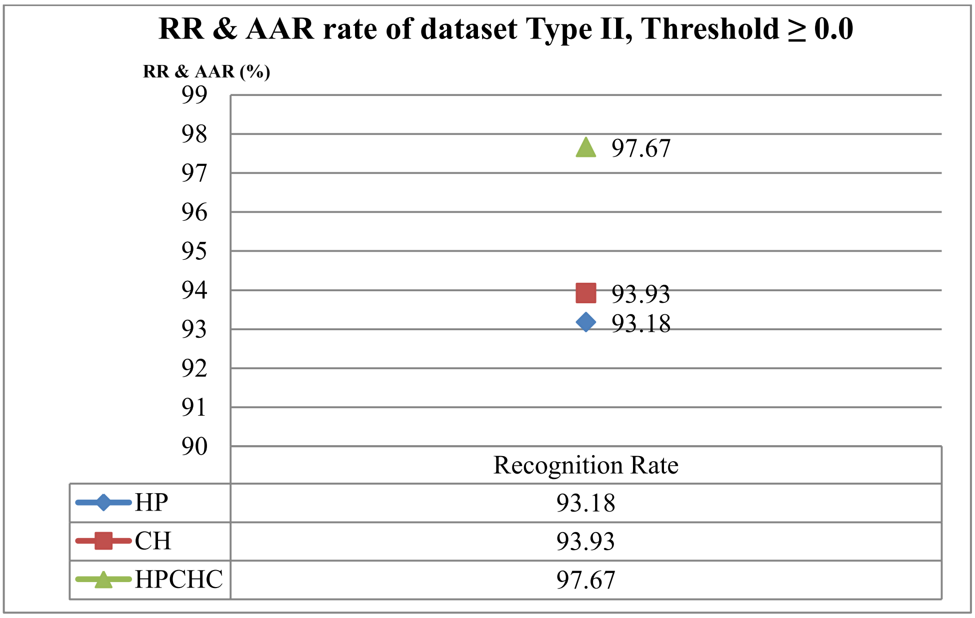


Figure 5.5: A graph illustrating the best Recognition Rate (RR) and Assessment Accuracy Rate (AAR) of each handwriting type of dataset Type II with threshold ≥ 0.0, employing ANNs as the classifier

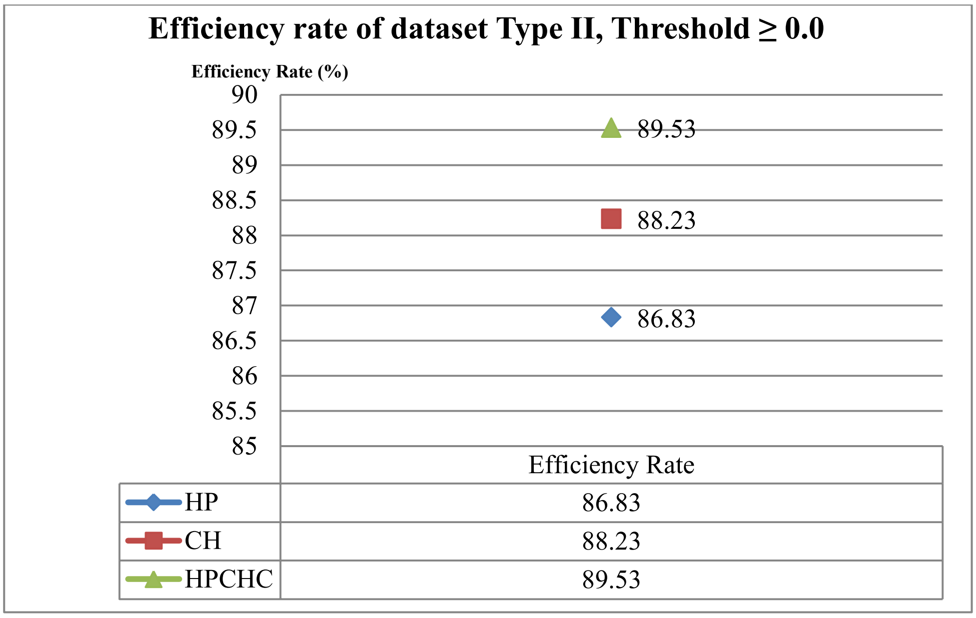


Figure 5.6: A graph illustrating the best efficiency rate of each handwriting type of dataset Type II with threshold ≥ 0.0, employing ANNs as the classifier

5.5.2.2 Dataset Type II Employing Threshold ≥ 0.0 Criterion

The results of dataset Type II which applied threshold ≥ 0.5 criterion are described as follows. From Table 5.11 and Figure 5.7, it can be seen that the highest recognition rate amongst the three handwriting types was 87.12% which was achieved from employing EGGF_D on FBC of HPCHC samples, while the second highest recognition rate of the HP samples of 85.61%, was obtained from employing MDF on FBC. The lowest recognition rate of the three handwriting types was 84.09% which was obtained from employing MDF to FBC or TI of CH samples. The differences between the highest and second highest, and the second highest and the lowest were 1.51% and 3.03% respectively. It can be concluded that similar to the highest recognition rate employing threshold ≥ 0.0 criterion, because the HPCHC contained samples from both HP and CH datasets, therefore, more samples were available for training and testing processes. As a result, the highest recognition rate of the three kinds of writing was achieved.

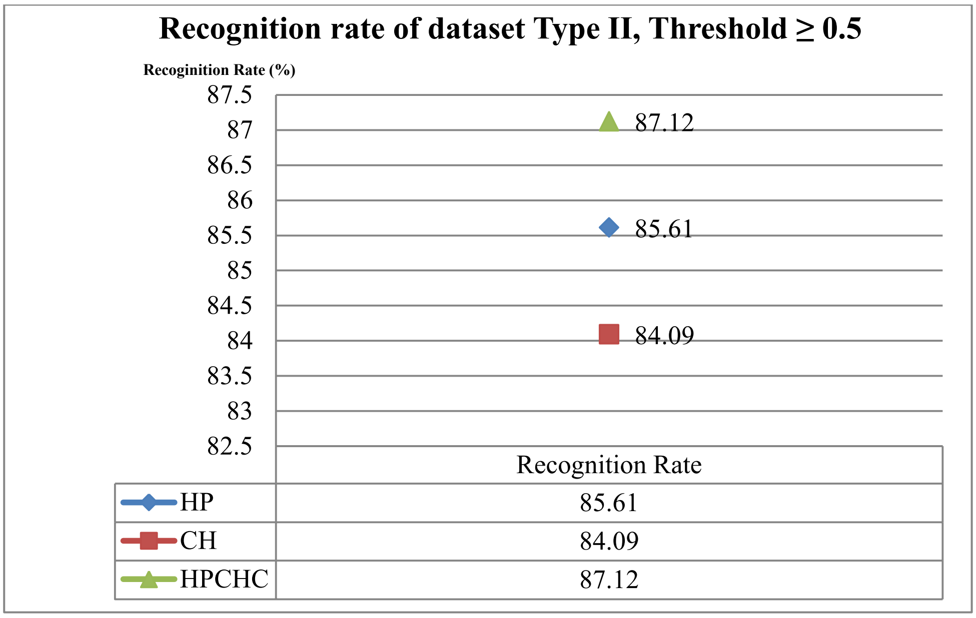


Figure 5.7: A graph illustrating the best recognition rate of each handwriting type of dataset Type II with threshold ≥ 0.5, employing ANNs as the classifier

Table 5.11: The best recognition rates of each handwriting type of dataset Type II with threshold ≥ 0.5

Handwriting Type	FET	Image Type	Recognition Rate (%)
HP	MDF	FBC	85.61
	MDF	FBC	
CH	MDF	TI	84.09
HPCHC	EGGF_D	FBC	87.12

The best assessment accuracy rates can be found in Table 5.12 and Figure 5.8 ranked as follows. The highest assessment accuracy of the three handwriting types was 96.09% which was attained from employing the MDF on FBC or TI of CH samples. The second highest assessment accuracy rate was 95.71% of HPCHC samples, achieved by employing MDF on TI, while the lowest assessment accuracy rate of the three was 94.75% obtained when employing WRGGF on FBC of HP samples.

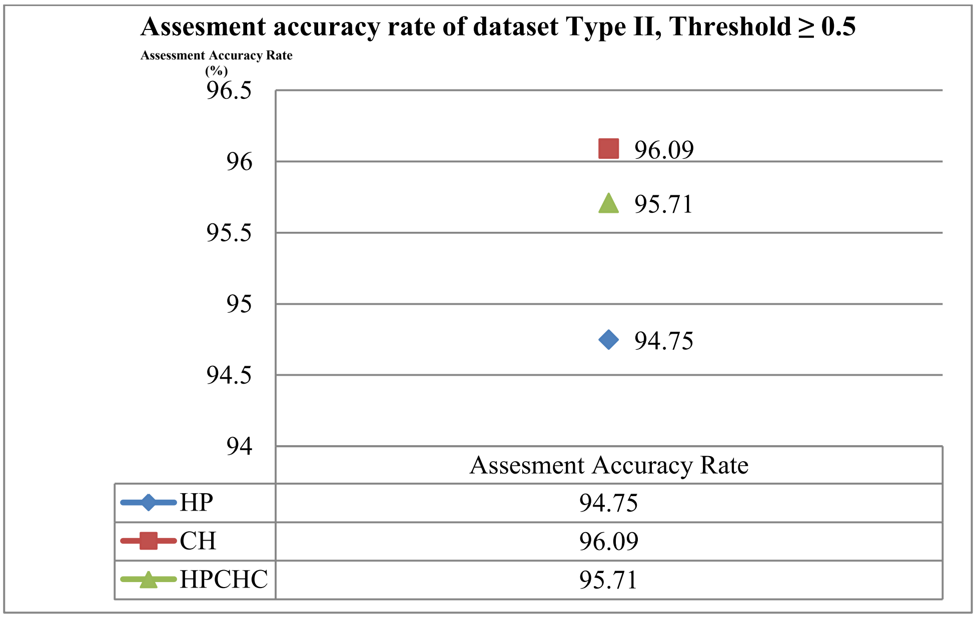


Figure 5.8: A graph illustrating the best assessment accuracy rate of each handwriting type of dataset Type II with threshold ≥ 0.5, employing ANNs as the classifier

Table 5.12: The best assessment accuracy rates of each handwriting type of dataset Type II with threshold ≥ 0.5

Handwriting Type	FET	Image Type	Assessment Accuracy Rate (%)
HP	WRGGF	FBC	94.75
	MDF	FBC	
CH	MDF	TI	96.09
HPCHC	MDF	TI	95.71

Table 5.13 and Figure 5.9 show that out of the three handwriting types, the highest efficiency rate was 81.66% which was attained from employing the proposed EMDF_A to FBC of HPCHC dataset. The second highest efficiency rate was obtained by employing MDF on FBC of HP dataset, and the lowest efficiency rate of the three handwriting type was 84.09% of the CH dataset employing MDF on either FBC or TI.

With dataset Type II, employing threshold ≥ 0.5 criterion, when compared to the recognition rates to each handwriting type assessment accuracy rate, it can be observed that all of the accuracy rates were higher the recognition rates. The assessment accuracy rates were higher than the recognition rates by 9.14%, 12%, and 8.59% for HP, CH, and HPCHC handwriting samples respectively. However, even though the assessment accuracy rates were higher than the recognition rates, the efficiency rates were found to be lower than both recognition and assessment accuracy rates. While recognition rates ranged between 84.09% – 87.12% and the assessment accuracy rates ranged between 94.75% – 96.09%, the efficiency rates only ranged between 74.99% – 81.66%. This can be explained by the fact that by employing threshold ≥ 0.5 criterion, less words were able to pass the threshold, and therefore less words were recognised. Also because of this reduced word recognition, even though the assessment accuracy rates were higher than the recognition rates, the efficiency rates were also low.

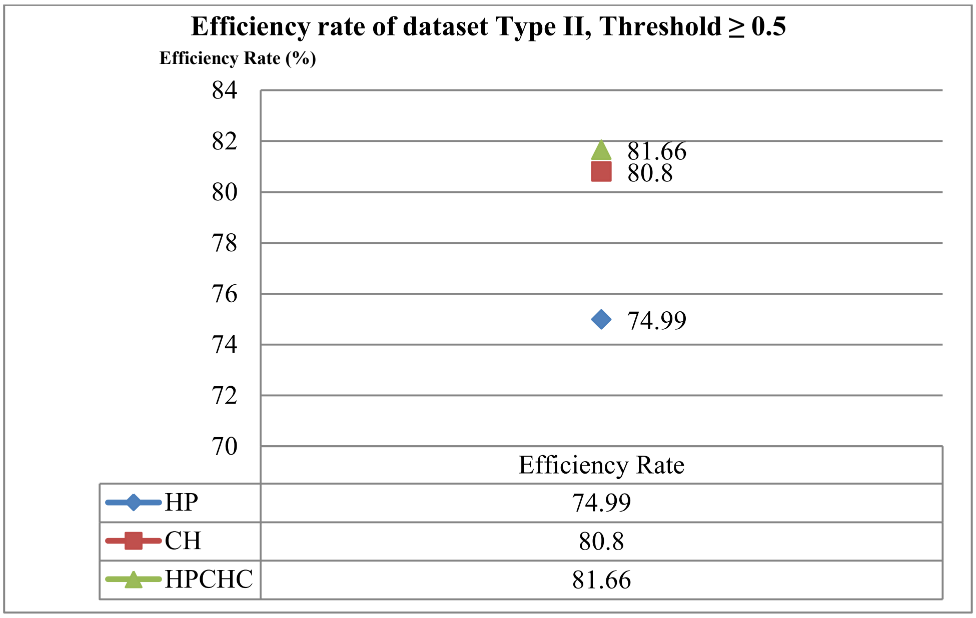


Figure 5.9: A graph illustrating the best efficiency rate of each handwriting type of dataset Type II with threshold ≥ 0.5, employing ANNs as the classifier

Table 5.13: The best efficiency rates of each handwriting type of dataset Type II with threshold ≥ 0.5

Handwriting Type	FET	Image Type	Efficiency Rate (%)
HP	MDF	FBC	74.99
	MDF	FBC	
CH	MDF	TI	80.80
HPCHC	EMDF_A	FBC	81.66

The graph in Figure 5.10 illustrates the best recognition comparison between threshold ≥ 0.0 criterion and threshold ≥ 0.5 criterion. As can be seen, the recognition rates of all handwriting types were higher when employing threshold ≥ 0.0 criterion. The better recognition rates of 10.55%, 9.84%, and 7.57% for HPCHC, CH, and HP handwriting samples, respectively, were observed.

An examination of the efficiency rates from the graph in Figure 5.11 reveals that the assessment accuracy rates of HP and CH handwriting samples had raised by 1.57% and 2.16% respectively, whilst the assessment accuracy for the HPCHC dataset dropped from 97.67% to 95.71% when threshold ≥ 0.5 criterion was employed. From these results it was determined that when employing threshold criteria in the future, the threshold ≥ 0.0 criterion may be utilised for HP and CH handwriting types, while the threshold ≥ 0.5 criterion may be applied to HPCHC handwriting types. However, to prove if this conclusion is valid, experiments on larger datasets of all handwriting types will need to be performed.

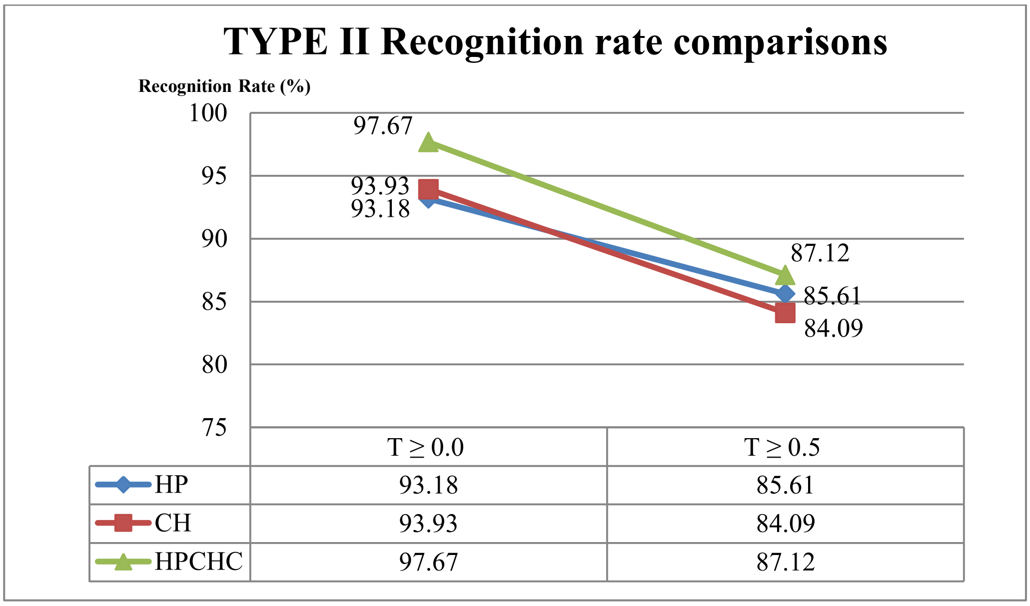


Figure 5.10: A graph illustrating the best recognition accuracy rate of each handwriting type of dataset Type I and II, employing ANNs as the classifier

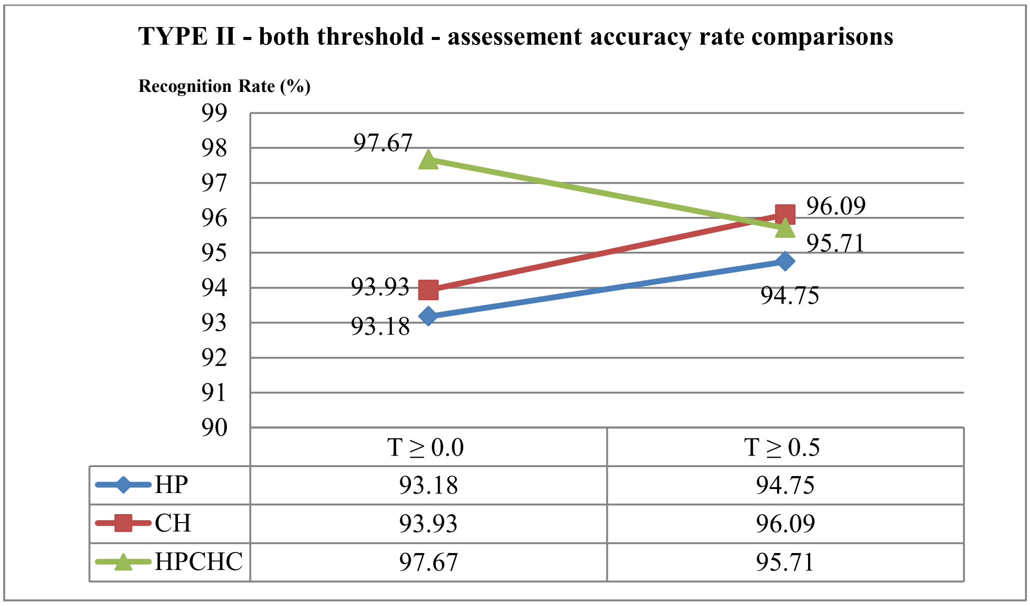
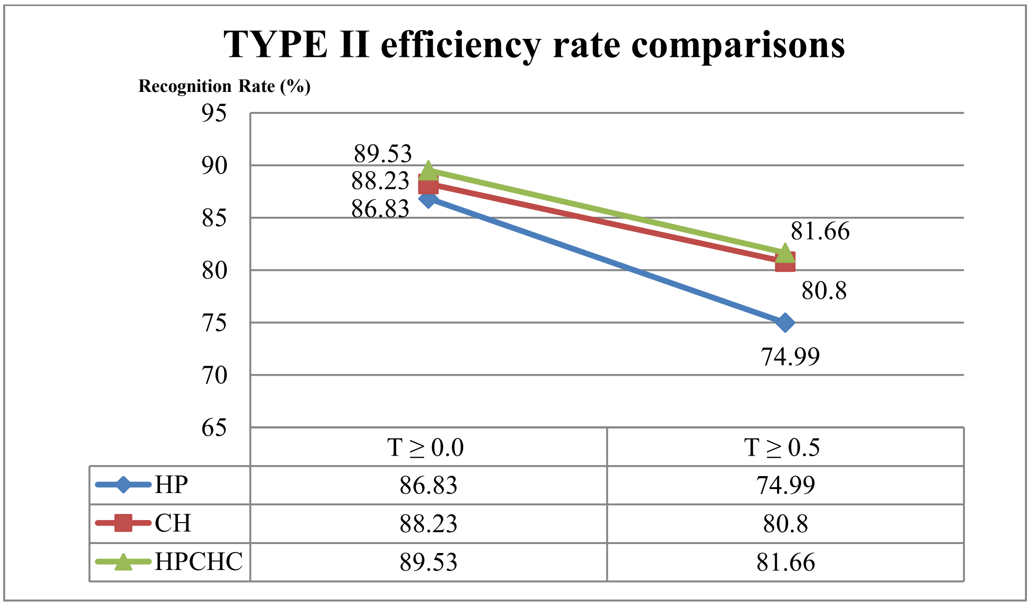
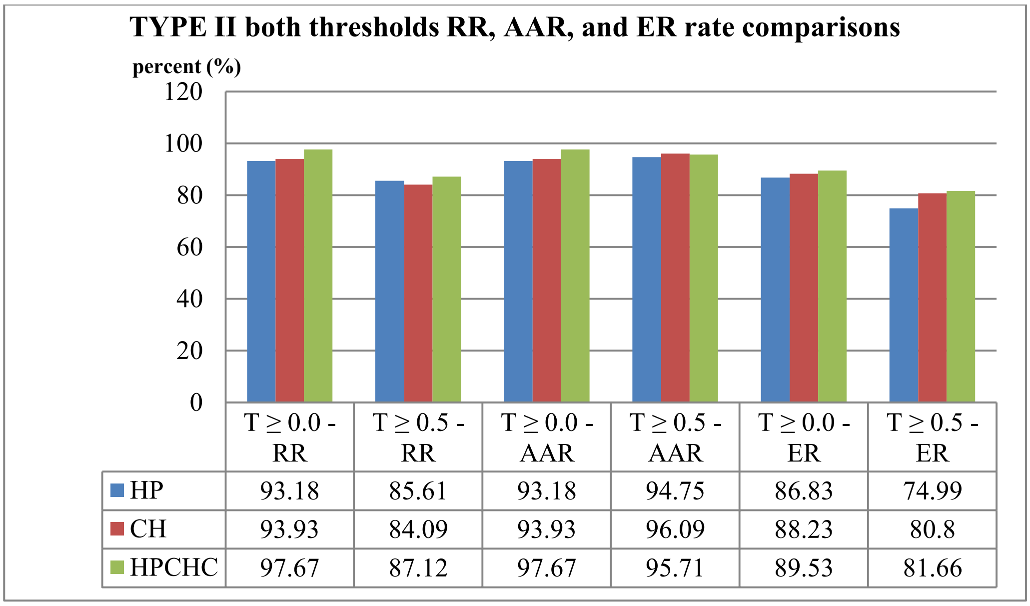


Figure 5.11: A graph illustrating the best assessment accuracy rate of each handwriting type of dataset Type I and II, employing ANNs as the classifier



	T ≥ 0.0	T ≥ 0.5
HP	86.83	74.99
CH	88.23	80.8
HPCHC	89.53	81.66

Figure 5.12: A graph illustrating the best efficiency rate of each handwriting type of dataset Type II, employing ANNs as the classifier



	T ≥ 0.0 - RR	T ≥ 0.5 - RR	T ≥ 0.0 - AAR	T ≥ 0.5 - AAR	T ≥ 0.0 - ER	T ≥ 0.5 - ER
HP	93.18	85.61	93.18	94.75	86.83	74.99
CH	93.93	84.09	93.93	96.09	88.23	80.8
HPCHC	97.67	87.12	97.67	95.71	89.53	81.66

Figure 5.13: A graph illustrating the best efficiency rate of each handwriting type of dataset Type II, employing ANNs as the classifier

From Figure 5.12, it can be seen that all efficiency rates achieved when employing threshold ≥ 0.0 criterion were higher than those obtained when threshold ≥ 0.5 criterion was employed. For HP dataset, the efficiency rate dropped as much as 11.84% (from 86.83% to 74.99%), 7.43% (88.23% to 80.80%) for the CH dataset, and 7.87% (89.53% to 81.66%) for the HPCHC dataset. The graph in Figure 5.13 illustrates the recognition, assessment accuracy, and efficiency rate comparisons between the rates obtained from threshold ≥ 0.0 and threshold ≥ 0.5 criteria.

In analysis of the Type II testing datasets, which were used to evaluate the SAAS, some factors that would affect the recognition, assessment accuracy, and efficiency rates were found.

A. Confusing words: Answers from the examinees including some confusing words which were hard for the classifier to recognise. A good example is the word ‘input’ and ‘output’, in that both of the words share a common component (put).

B. Poor writing: Some of the handwriting samples were hard to read, even for the human assessor as the students may have been writing under stress and writing so fast that it reduced legibility. This factor has a direct effect on the recognition rate.

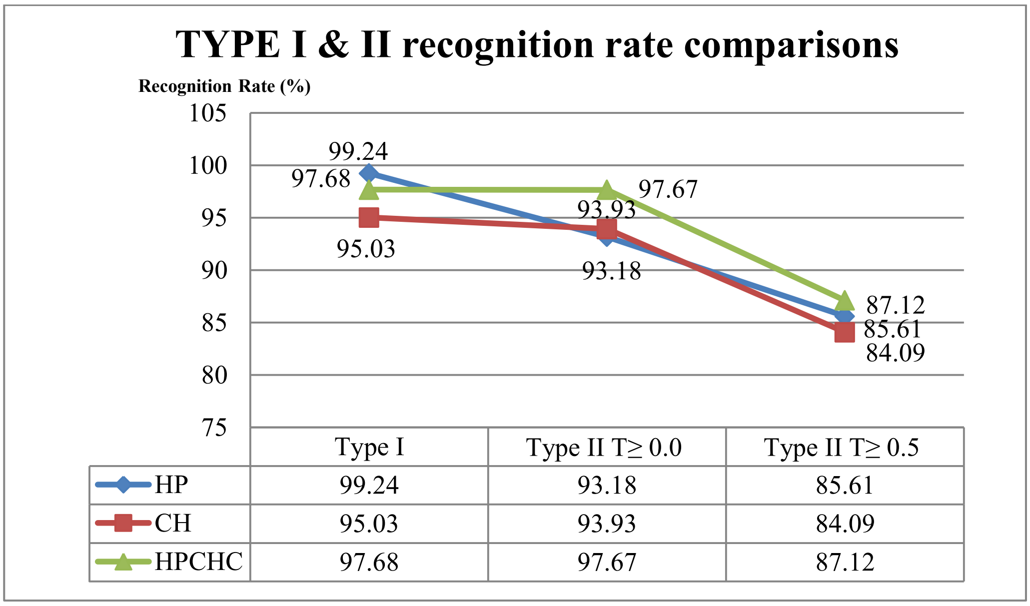
C. Poor resolution of images: The resolution of the scanned examination papers for the datasets was quite low. A lot of noise was found.

D. Poor positioning of words in word matrices: The images used in all experiments had not been slant corrected. The images were used as they were, except for some images where skew normalisation needed to be performed.

It is important to note that even though some feature extraction techniques gave better recognition rates and efficiency when compared to the others, the most important consideration when marking examination papers is to mark them with the highest assessment accuracy possible. In this SAAS application, therefore, the assessment accuracy rate is considered the most important rate when compared to recognition rate because it does not matter how high recognition rates are. If the accuracy rates are low, the system is still regarded as impractical and therefore unsuitable to be used in a real world situation.

5.5.3 Dataset Type I and II Result Comparison and Analysis

This sub-section presents dataset Type I and II results comparison and analysis which are described as follows. The graph in Figure 5.14 illustrates that the highest recognition rates were obtained when dataset Type I was employed. This was expected as dataset Type I contained only correctly spelt correct words (answers to the questions) which meant that the testing dataset also only contained correctly spelt correct words. Because of the aforementioned reason that only the intra-class difference problem had to be dealt with, the highest recognition rates of all datatypes (Type I, Type II $t \geq 0.0$, Type II $t \geq 0.5$) were achieved. For the highest recognition rates of datasets Type II, because the dataset contained some misspelt or wrong answers as well as correctly spelt correct answers in the testing dataset, the highest recognition rates obtained did not outperform dataset Type I recognition rates.



	Type I	Type II T≥ 0.0	Type II T≥ 0.5
HP	99.24	93.18	85.61
CH	95.03	93.93	84.09
HPCHC	97.68	97.67	87.12

Figure 5.14: A graph illustrating the best recognition rate of each handwriting type of dataset Type I and II, employing ANNs as the classifier

5.6 SAAS Comparison to other automatic assessment systems Found in The Literature

The main piece of research found in the literature on off-line automated assessment systems, which is similar to that proposed here, was that of Allan [9]. However, the existing automated assessment systems and the system proposed here are quite different. It is also hard to compare the classification and accuracy rates due to the differences between the nature of the systems and the criteria used to determine such rates. However, the features of both systems are discussed below.

The comparisons between the systems include the dataset, feature extraction, assessment type, and recognition and accuracy rate. The discussion is mainly based around existing systems where their experiments were conducted in 2001 [10]. Some features of the other existing systems are also discussed here.

A. Datasets: The datasets used in the literature were quite varied. These included handwriting from children to adults, depending on the experiments conducted. The dataset from the literature, which is discussed here, was collected from 52 1st year computing students. The examination paper was made up of eight multiple choice questions (MCQ). Each question had 3 responses to choose from. The examinees were required to write the answer from the 3 responses available.

The datasets used in the present study included 104 examination papers, which were larger than the existing datasets. The examination papers were divided into three categories: HP, CH, and HPCHC. The examination was a short answer handwritten examination which contained 10 questions.

B. Feature Extraction Techniques: The existing system used a structure feature extraction technique with a holistic HVBC recogniser [72]. The holistic recogniser works by recognising the shape of the word from features extracted from the whole word image. The cursive script recognition technique was also used in the other experiment.

The current proposed system employed eighteen feature extraction techniques which were 1) MDF; enhanced MDF, namely, 2) EMDF_A; 3) EMDF_B; 4) EMDF_C; and 5) EMDF_D; 6) GGF; enhanced GGF, namely, 7) EGGF_A; 8) EGGF_B; 9) EGGF_C; 10) EGGF_D; 11) WRLGGF; enhanced WRLGGF, namely, 12) EWRLGGF_A; 13) EWRLGGF_B; 14) EWRLGGF_C; 15) EWRLGGF_D; 16) WRGGF; 17) LGGF; and 18) MWRLGGF. In this study, all of the feature extraction techniques also extract features from the whole word.

C. Assessment: There were a few approaches employed in the previous research namely: 1) conventional lexical approach which all written words held within the lexicon; 2) specific word assessment technique which exploits the nature of the question and answer medium by only comparing the input pattern to the template of the correct answer for each question; 3) contextual word bridges approach. Some of the approaches were employed in children's handwritten responses assessment systems; however, these will not be discussed here. The assessment reported in existing research, which was used in the system comparison, was performed on multiple choice questions (MCQs) employing a contextual word bridges approach. Each response consisted of a selection of three words. These words were used so that bridges between each word were created. The number of bridges was counted to consider the recognition responses which are valid, possible and invalid. For example, a valid bridge can be found when two words are correctly recognised in the correct order. The answers used in the existing system were restricted to the words provided. The results that obtained higher recognition rates are due to the responses and the nature of the restricted answers from the examinees that were known prior to the assessment process.

The proposed system does not have restrictions or any constraints in the question answering process. Even though the questions are closed ended, the examinees are allowed to write the answer freely. The proposed system also has the ability to assess the examination paper based on the quality of the answer. The assessment criterion enables the system to be used in a real world situation, where marking the answers is mainly based on their quality.

D. Recognition and Accuracy Rates: The existing system [9] was reported to be able to achieve the assessment yield range from a recognition rate of 54% with 99% assessment accuracy to a 44% recognition rate with a 100% assessment accuracy depending on the constraints, such as lexicon and bridges between the lexicons, as well as the response history applied. The main system which is discussed here correctly scored 54% of all responses with an accuracy rate of 99%. The rates were attained by using bridges between each word, which may be considered a constraint.

The proposed system using HP words achieved recognition and assessment accurate rates of 93.18% when threshold ≥ 0.0 was employed, or 85.61% recognition rate with 94.75% assessment accuracy rate when threshold ≥ 0.0 was employed. The CH words attained recognition and assessment accurate rates of 93.93% when threshold ≥ 0.0 was employed and the recognition rate of 84.09% with 96.09% assessment accuracy rate when threshold ≥ 0.0 was employed. For the HPCHC words, the best recognition and assessment accurate rate of 97.67% was achieved when threshold ≥ 0.0 was employed, and the recognition rate of 87.12% with 95.71% assessment accuracy rate was obtained when threshold ≥ 0.0 was employed. The recognition and assessment accuracy rates were obtained without applying any strict constraints (e.g. writing instruments were not restricted) when compared to the existing systems.

Chapter 6

CONCLUSIONS AND FUTURE RESEARCH

The research proposed in this thesis has investigated successful feature extraction techniques for the problem word recognition of an off-line short answer automated assessment system with student identification system. The main objectives were to investigate the performance of the Modified Direction Feature (MDF) and Gaussian Grid Feature (GGF) extraction techniques, and to propose more efficient feature extraction techniques. Both of the objectives have been successfully achieved. Various experimental methods and settings have been investigated and implemented successfully in the proposed bilingual name components identification and short answer words automatic assessment systems.

Investigation on the contour deduction was performed. The experiments were performed by employing upper and lower contour and loops (Three Images - TI), upper and lower contours (ULC), and Full Boundary Contour (FBC) in the feature extraction process. The feature extraction techniques were performed on the three images without extracting from the full boundary images.

New datasets of English and Thai (bilingual) name components and short answer words were collected in order to be employed in the experiments of the proposed systems, as there was no dataset from the existing systems available. The results of these experiments presented in Chapter 4 and 5 evidently indicate that the newly proposed Water Reservoir, Loop, and Modified Direction Feature (WRLMDF) outperformed the original MDF a number of experiments for examples, in 3,940 Thai name component samples where SVMs was employed as classifier by 1.19% when utilising the WRLMDF on FBC and 2.6% on TI respectively. More recognition rates have improved English name component of 3,940 samples where SVMs was employed as classifier by 1.37% when utilising the WRLMDF on FBC and 3.75% on TI respectively. For the bilingual name component dataset, the recognition rates have improved by employing SVMs as classifier by as much as 1.74% when utilising the WRLMDF on FBC and 4.19% on TI respectively.

The proposed Water Reservoir, Loop, and Gaussian Grid Feature (WRLGGF) achieved the best recognition rate out of the following system compared to the original MDF and GGF feature extraction techniques. WRLGGF achieved the highest recognition rates of 99.52%, 97.49%, 95.94%, 99.25% and 95.99% of 1,040 sample ESIS, 3,940 sample ESIS, 3,940 sample TSIS, 2,060 sample BSIS, and 7,880 sample BSIS, respectively. The proposed WRLGGF also yielded the best recognition rate of 94.75% when employed on the hand-printed short answer dataset employing threshold ≥ 0.5 criterion (referring to Chapter 3 and 4). It can be noted that by employing the original GGF on ULC of dataset Type I (see Chapter 3 and 4), the best recognition rate (out of all other feature extraction techniques employed) of 99.24% was achieved.

The proposed enhanced MDF, GGF, WRLGGF extraction techniques, as can be seen from the experiment results in Chapter 4, have outperformed every original feature extraction technique (MDF, GGF, and WRLGGF). For short answer Type I dataset employing ANNs as the classifier, the better recognition rate by 1.14% was obtained for enhanced MDF, GGF, and WRLGGF compared to the original MDF, GGF and WRLGGF, respectively. For the short answer Type I dataset employing SVMs as the classifier, the better recognition rates of 0.96%, 0.32%, and 0.16%; these rates were obtained for enhanced MDF, GGF, and WRLGGF compared to the original MDF, GGF and WRLGGF, respectively.

With short answer Type II dataset, the better recognition rates of 1.89%, 6.06%, and 1.89% were obtained when employing enhanced MDF, GGF, and WRLGGF compared to the original MDF, GGF and WRLGGF, respectively. The assessment accuracy rates of the Type II dataset were also improved by 0.89%, 1%, and 1.89%; these rates were obtained for enhanced MDF, GGF, and WRLGGF compared to the original MDF, GGF and WRLGGF, respectively. Similar results were achieved for short answer Type II dataset efficiency rates. The improved efficiency rates of 2.54%, 3.95%, and 1.79% were attained when employing the enhanced MDF, GGF, and WRLGGF compared to the efficiency rates of the original MDF, GGF and WRLGGF, respectively.

The originality of this research includes the proposed complete automatic short answer assessment system with student identification system, which consists of a SAAS and SIS as well as newly proposed WRLGGF, WRLMDF and enhanced MDF, GGF, and WRLGGF feature extraction techniques which were able to achieve better recognition, assessment accuracy, and efficiency rates when compared to the original feature extraction techniques. While some of the proposed feature extraction techniques were able to attain substantial improvements, some were only able to obtain smaller improvements. The feature extraction techniques proposed in this thesis were developed to be employed in the proposed SAAS and SIS. As a result even though some of the proposed feature extraction techniques could not achieve substantial improvement when compared to their numbers of features (feature vector sizes), they were still encouraging results. For any increment in recognition, assessment accuracy and efficiency rate (especially in assessment accuracy rate), this can be considered significant for SAAS as marking examinations incorrectly could result in a student failing the subject, and therefore worth the trade-off between feature vector sizes and the improvements attained.

Also, new datasets were created for the experiments. The proposed SAAS system was able to recognise and mark examination papers, together with the SIS being able to identify students from their name components successfully. The results produced by the proposed system were encouraging.

6.1 Future Research

A substantial number of experiments have been conducted in the present research; therefore many notable experimental results and useful observations have been recorded and documented. Newly collected Thai and English (bilingual) name component datasets, and short answer word dataset obtained from examination papers are available to the research community which can be obtained upon request. These factors are, therefore, enabling further investigation in the area of off-line handwriting recognition, especially for student identification and short answer examination marking systems/applications, so further improvements can be made.

Further employment of feature extraction techniques and classifiers can be taken into consideration in order to improve the results for future research. Additional feature extraction techniques can be considered for recognition, accuracy and efficiency rates, and usability improvement. For future research, enhanced feature extraction techniques may extended by including extracting other important features such as background and foreground components, SURF, or rotation variant-based features from images. Words and name components can be found in noisy environments; also they can be found to be incomplete or heavily misspelt, heavily skewed or/and slanted, crossed out, hand stroked through, or scribbled writing, and therefore, need to be considered for future work.

Employment of other classifiers, especially Hidden Markov Models (HMMs) which are able to recognise handwritten words without the explicit segmentation, is other important factor which may help to improve the performance of the systems. Further investigation in employing other classifiers or hybrid classifiers should therefore be carried out in order to increase the aforementioned rates and usability.

In the near future, student identification systems can be extended to student identification and verification systems where biometrics are involved which will expand further the usability of the student identification system. The short answer automated assessment system can also be extended into long answer or essay marking systems. Last but not least, larger size datasets together with more languages/scripts can also be considered in the future work.

Bibliography

1. Coulmas, F., The writing systems of the world, 1991, Blackwell.

2. Plamondon, R. and S.N. Srihari. Online and off-line handwriting recognition: a comprehensive survey. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 22, p. 63-84, 2000.

3. Arica, N. and F.T. Yarman-Vural, Optical character recognition for cursive handwriting. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 24, p. 801-813, 2002.

4. L O'Gorman, R.K. Executive briefing: document image analysis, 1997.

5. Tauschek, G. U.S. Patent 2026329, Reading Machine, 1935.

6. Handel, P. U.S. Patent 1915993, Statistical machine, 1933.

7. Mori, S., C.Y. Suen, and K. Yamamoto. Historical review of OCR research and development. In Proceedings of the IEEE, volume 80, p. 1029-1058, 1992.

8. Blumenstein, M. Intelligent Techniques for Handwriting Recognition. PhD thesis, Griffith University, Gold Coast, 2000.

9. Allan, J. Automated Assessment of Handwritten Scripts. PhD thesis, The Nottingham Trent University, 2004.

10. Allan, J., T. Allen, N. Sherkat, and P. Halstead. Automated assessment: it's assessment Jim but not as we know it. In Proceeding of Sixth International Conference on Document Analysis and Recognition, p. 926-930, 2001.

11. Allan, J., T. Allen, and N. Sherkat. Automated assessment: how confident are we? In Proceedings of the Eighth International Workshop on the Frontiers in Handwriting Recognition, p. 419-423, 2002.

12. Allan, J., T. Allen, and N. Sherkat. Confident assessment of children's handwritten responses. In Proceedings of Eighth International Workshop on Frontiers in Handwriting Recognition, 2002.

13. Srihari, S., Collins, J., Srihari, R., Srinivasan, H., Shetty, S., Brutt-Griffler, J. Automatic scoring of short handwritten essays in reading comprehension tests. Artificial Intelligence, volume 172, p. 300-324, 2008.

14. Ghosh, S. and S.S. Fatima. Design of an Automated Essay Grading (AEG) system in Indian context. In Proceedings of the 2008 IEEE Conference Region 10 TENCON, 2008.

15. Box, I., Assessing the assessment: an empirical study of an information systems development subject, In Proceedings of the fifth Australasian conference on Computing education, Australian Computer Society, Inc.: Adelaide, Australia. p. 149-158, 2003.

16. Tappert, C.C., C.Y. Suen, and T. Wakahara,. The state of the art in online handwriting recognition. In IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 12, p. 787-808,1990.

17. Ahmad, A.R., C. Viard-Gaudin, and M. Khalid. Lexicon-Based Word Recognition Using Support Vector Machine and Hidden Markov Model. In 10th International Conference on Document Analysis and Recognition (ICDAR '09), p. 161-165, 2009.

18. Caesar, T., J.M. Gloger, and E. Mandler. Preprocessing and feature extraction for a handwriting recognition system. In Proceedings of the Second International Conference on Document Analysis and Recognition, p. 408-411, 1993.

19. Liwicki M., G.A., Bunke H., Schmidhuber J., A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In Proceedings of the 9th International Conference on Document Analysis and Recognition, p. 367–371, 2007.

20. Radhika, K.R., G.N. Sekhar, and M.K. Venkatesha. Pattern recognition techniques in on-line hand written signature verification - A survey. In Proceedings of the International Conference on Multimedia Computing and Systems (ICMCS '09), p. 216-221, 2009.

21. Namboodiri, A.M. and A.K. Jain, Online handwritten script recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 26, p. 124-130, 2004.

22. Koerich, A.L., R. Sabourin, and C.Y. Suen, Large vocabulary off-line handwriting recognition: A survey. Pattern Analysis & Applications, volume 6(2), p. 97-121, 2003.

23. Govindaraju, V. and R.K. Krishnamurthy, Holistic handwritten word recognition using temporal features derived from off-line images. Pattern Recognition Letters, volume 17, p. 537-540, 1996.

24. Kulkarni, A.D., Computer vision and fuzzy-neural systems. Upper Saddle River, NJ Prentice Hall PTR, 2001.

25. Rehman, A. and T. Saba, Neural networks for document image preprocessing: state of the art. Artificial Intelligence Review, volume 42, p. 253-273, 2014.

26. Kato, M., T. Kosaka, A. Ito, S. Makino, An Algorithm for Fast Calculation of Back-off N-gram Probabilities with Unigram Rescaling. IAENG International Journal of Computer Science, 2009.

27. Trier, O.D. and A.K. Jain, Goal-directed evaluation of binarization methods. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume17, p. 1191-1201, 1995.

28. Vargas, J.F., Ferrer, M. A., Travieso, C. M., Alonso, J. B. Off-line Signature Verification System Performance against Image Acquisition Resolution. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR' 07), p. 834-838, 2007.

29. Otsu, N., A Threshold Selection Method from Gray-Level Histograms. In Proceedings of the IEEE Transactions on Systems, Man, and Cybernetics (SMC '09), p. 62-66, 1979.

30. Touj, S.M., N.E. Ben Amara, and H. Amiri. A hybrid approach for off-line Arabic handwriting recognition based on a Planar Hidden Markov modeling. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR '07), p. 964-968, 2007.

31. Gilloux, M., B. Lemarie, and M. Leroux. A hybrid radial basis function network/hidden Markov model handwritten word recognition system. In Proceedings of the Third International Conference on Document Analysis and Recognition, p. 394-397, 1995.

32. Senior, A.W. and A.J. Robinson, An off-line cursive handwriting recognition system. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 20, p. 309-321, 1998.

33. Due Trier, Ø., A.K. Jain, and T. Taxt, Feature extraction methods for character recognition-A survey. Pattern Recognition, volume 29, p. 641-662, 1996.

34. Parvez, M.T. and S.A. Mahmoud, Offline Arabic handwritten text recognition: A Survey. ACM Computer Survey, volume 45(2): p. 1-35, 2013.

35. Mozaffari, S., et al. A Comprehensive Isolated Farsi/Arabic Character Database for Handwritten OCR Research. In Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule (France), 2006.

36. Trier, O.D. and T. Taxt, Evaluation of binarization methods for document images. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 17(3), p. 312-315 1995.

37. Pratikakis, I., B. Gatos, and K. Ntirogiannis. ICDAR 2013 Document Image Binarization Contest (DIBCO 2013). In Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR '13), 2013.

38. Kittler, J. and J. Illingworth, Minimum error thresholding. Pattern Recognition, volume 19(1), p. 41-47, 1986.

39. Kapur, J.N., P.K. Sahoo, and A.K.C. Wong, A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision, Graphics, and Image Processing, volume 29(3), p. 273-285, 1985.

40. Liu, C.-L. and C.Y. Suen, A new benchmark on the recognition of handwritten Bangla and Farsi numeral characters. Pattern Recognition, 42(12), p. 3287-3295, 2009.

41. Sagheer, M.W., et al. Holistic Urdu Handwritten Word Recognition Using Support Vector Machine. In Proceedings of the 20th International Conference on Pattern Recognition (ICPR '20), p. 1900-1903, 2010.

42. Ferrer, M.A., et al., Robustness of Offline Signature Verification Based on Gray Level Features. IEEE Transactions on Information Forensics and Security, volume 7(3), p. 966-977, 2012.

43. Vargas, J.F., et al. Offline Signature Verification Based on Pseudo-Cepstral Coefficients. In Proceedings of the 10th International Conference on Document Analysis and Recognition (ICDAR '09), p. 126-130, 2009.

44. Huang, S., M. Ahmadi, and M.A. Sid-Ahmed, A hidden Markov model-based character extraction method. Pattern Recognition, volume 41(9), p. 2890-2900, 2008.

45. Huang, D.-Y. and C.-H. Wang, Optimal multi-level thresholding using a two-stage Otsu optimization approach. Pattern Recognition Letters, volume 30, p. 275-284, 2009.

46. Suen, C.Y., M. Berthod, and S. Mori, Automatic recognition of handprinted characters: The state of the art. In Proceedings of the IEEE, volume 68(4), p. 469-487, 1980.

47. Methasate, I. and S. Sae-tang. The clustering technique for Thai handwritten recognition. In Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), p. 450-454, 2004.

48. Chen, M.Y., A. Kundu, and S.N. Srihari, Variable duration hidden Markov model and morphological segmentation for handwritten word recognition. IEEE Transactions on Image Processing, volume 4(12), p. 1675-1688, 1995.

49. Shi, Z. and V. Govindaraju, Character image enhancement by selective region-growing. Pattern Recognition Letters, volume 17(5), p. 523-527, 1996.

50. Siyuan, C. and S. Sargur. Use of exterior contours and shape features in off-line signature verification. In Proceedings of the Eighth International Conference on Document Analysis and Recognition, p. 1280-1284, 2005.

51. Srikantan, G., L. Dar-Shyang, and J.T. Favata. Comparison of normalization methods for character recognition. In Proceedings of the 3rd International Conference on Document Analysis and Recognition, p. 719-722, 1995.

52. A.Vinciarelli, J.L., Off-line cursive script recognition based on continuous density HMM., 2000.

53. Bozinovic, R.M. and S.N. Srihari, Off-line cursive script word recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 11(1), p. 68-83, 1989.

54. Chow, Y.-L. and R. Schwartz, The N-Best algorithm: an efficient procedure for finding top N sentence hypotheses, In Proceedings of the workshop on Speech and Natural Language, Association for Computational Linguistics: Cape Cod, Massachusetts, p. 199-202, 1989.

55. A. Brakensiek, J.R., A. Kosmala, G. Rigoll, Off-Line Handwriting Recognition Using Various Hybrid Modeling Techniques and Character N-Grams. 2000.

56. Bunke, H., M. Roth, and E.G. Schukat-Talamazzini, Off-line cursive handwriting recognition using hidden markov models. Pattern Recognition, volume 28(9), p. 1399-1413, 1995.

57. Bunke, H. Recognition of cursive Roman handwriting: past, present and future. In Proceedings of the Seventh International Conference on Document Analysis and Recognition, p. 448-459, 2003.

58. Madhvanath, S. and V. Govindaraju, Local reference lines for handwritten phrase recognition. Pattern Recognition, volume 32(12), p. 2021-2028, 1999.

59. Dong, L., et al. Applied some new features in off-line recognition of totally unconstrained handwritten numerals using neural network. In Proceedings of the IEEE

International Conference on Intelligent Processing Systems (ICIPS '97), p. 392-395, 1997.

60. Ha, T. and H. Bunke, Image Processing Methods for Document Image Analysis. Handbook of Character Recognition and Document Image Analysis. World Scientific, 1997.

61. Parker, J.R., Practical Computer Vision using C. 1994, New York, NY: John Wiley and Sons.

62. Tai-Ping, Z., et al. Signature envelope curvature descriptor for offline signature verification. In Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR '07), p. 1262-1266, 2007.

63. Blumenstein, M., X.Y. Liu, and B. Verma, An investigation of the modified direction feature for cursive character recognition. Pattern Recognition, volume 40(2): p. 376-388, 2007.

64. Kessentini, Y., T. Paquet, and A. Ben Hamadou, Off-line handwritten word recognition using multi-stream hidden Markov models. Pattern Recognition Letters, volume 31(1): p. 60-70, 2010.

65. Methasate, I., et al. The feature combination technique for off-line Thai character recognition system. In Proceedings of the Eighth International Conference on Document Analysis and Recognition, p. 1006-1009, 2005.

66. de S. Britto, A., et al. Foreground and background information in an HMM-based method for recognition of isolated characters and numeral strings. In Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), p. 371-376, 2004.

67. Kundu, A. and P. Bahl. Recognition of handwritten script: a hidden Markov model based approach. In Proceedings of the 1988 International Conference on Acoustics, Speech, and Signal Processing (ICASSP-88), p. 928-931, 1988.

68. Yi-Kai, C. and W. Jhing-Fa, Segmentation of single- or multiple-touching handwritten numeral string using background and foreground analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 22(11), p. 1304-1317, 2000.

69. Sherkat, N. and T.J. Allen. Whole word recognition in facsimile images. In Proceedings of the Fifth International Conference on Document Analysis and Recognition, p. 547-550, 1999.

70. Raza, G., et al. Recognition of facsimile documents using a database of robust features. In Proceedings of the Fourth International Conference on Document Analysis and Recognition, p. 444-448, 1997.

71. Sherkat, N. Influence of zoning on whole word recognition. in Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on. 2005.

72. Sherkat, N., R.J. Whitrow, and R.G. Evans. Wholistic recognition of handwriting using structural features. In IEE Colloquium on Document Image Processing and Multimedia, p. 12/1-12/4, 1999.

73. Blumenstein, M., B. Verma, and B. H., A Novel Feature Extraction Technique for the Recognition of Segmented Handwritten Characters. In Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR '03), p. 137-141, Scotland, 2003.

74. Verma, B., M. Blumenstein, and M. Ghosh, A novel approach for structural feature extraction: Contour vs. direction. Pattern Recognition Letters, volume 25(9): p. 975-988, 2004.

75. Liu, C.-L., et al., Handwritten digit recognition: investigation of normalization and feature extraction techniques. Pattern Recognition, volume 37(2), p. 265-279, 2004.

76. Nishimura, H., et al. Off-line character recognition using HMM by multiple directional feature extraction and voting with bagging algorithm. In Proceedings of the Fifth International Conference on Document Analysis and Recognition (ICDAR '99), p. 49-52, 1999.

77. Gader, P.D., M. Mohamed, and C. Jung-Hsien, Handwritten word recognition with character and inter-character neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, volume 27(1), p. 158-164, 1997.

78. Verma, B., A Contour Code Feature Based Segmentation For Handwriting Recognition. In Proceedings of the Seventh International Conference on Document Analysis and Recognition, p. 1203 – 1207, 2003.

79. Mohamed, M. and P. Gader, Handwritten word recognition using segmentation-free hidden Markov modeling and segmentation-based dynamic programming techniques. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 18(5), p. 548-554, 1996.

80. Blumenstein, M., X.Y. Liu, and B. Verma. A modified direction feature for cursive character recognition. In Proceedings of the IEEE International Joint Conference on Neural Networks, p. 2983-2987, 2004.

81. Vu, N., et al. Performance Analysis of the Gradient Feature and the Modified Direction Feature for Off-line Signature Verification. In Proceedings of the 2010 International Conference on Frontiers in Handwriting Recognition (ICFHR), p. 303-307, 2010.

82. Vu, N., et al. Off-line Signature Verification Using Enhanced Modified Direction Features in Conjunction with Neural Classifiers and Support Vector Machines. In Proceedings of the Ninth International Conference on Document Analysis and Recognition, p. 734-738, 2007.

83. Green, S., et al. Investigation of a Classification-based Technique to Detect Illicit Objects for Aviation Security. In Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, p. 825-830, Austria, 2005

84. Green, S., et al. The detection and quantification of persons in cluttered beach scenes using neural network-based classification. In Proceedings of the Sixth International Conference on Computational Intelligence and Multimedia Applications, p. 303-308, 2005.

85. Green, S. and M. Blumenstein. An Exhaustive Search Strategy for Detecting Persons in Beach Scenes using Digital Video Imagery and Neural Network-based Classification. In Proceedings of the International Joint Conference on Neural Networks (IJCNN '06), p. 2439-2444, 2006.

86. Vu, N. and M. Blumenstein. An Application of the 2D Gaussian Filter for Enhancing Feature Extraction in Off-line Signature Verification In Proceedings of the 2011 International Conference on Document Analysis and Recognition, p. 339-343, 2011.

87. Procter, S., J. Illingworth, and F. Mokhtarian, Cursive handwriting recognition using hidden Markov models and a lexicon-driven level building algorithm. IEE Proceedings on Vision, Image and Signal Processing, volume 147, p. 332-339, 2000.

88. Pal, U. and B. Chaudhuri. Automatic Recognition of Unconstrained Off-Line Bangla Handwritten Numerals, Advances in Multimodal Interfaces — ICMI 2000, Springer Berlin / Heidelberg. p. 371-378, 2000.

89. Partha Pratim Roy, U.P., Josep Lladós. Morphology Based Handwritten Line Segmentation Using Foreground and Background Information. In the Proceedings of

the 2008 International Conference on Frontiers in Handwriting Recognition (ICFHR ‘08), p. 241–246, 2008.

90. Roy, P.P., et al. Multi-Oriented English Text Line Extraction Using Background and Foreground Information. In the Proceedings of the Eighth IAPR International Workshop on Document Analysis Systems (DAS '08), p. 315-322, 2008.

91. Doetsch, P., et al. Comparison of Bernoulli and Gaussian HMMs Using a Vertical Repositioning Technique for Off-Line Handwriting Recognition. In the Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition (ICFHR ‘12), p. 3-7, 2012.

92. Pfister, M., S. Behnke, and R. Rojas, Recognition of Handwritten ZIP Codes in a Real—World Non-Standard-Letter Sorting System. Applied Intelligence, volume 12, p. 95-115, 2000.

93. Kitadai, A. and M. Nakagawa. A learning algorithm for structured character pattern representation used in online recognition of handwritten Japanese characters. In the Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition, p. 163-168, 2002.

94. M. Blumenstein, C.K.C., and X.Y. Liu. New Preprocessing Techniques for Handwritten Word Recognition. In the Proceedings of the Second IASTED International Conference on Visualization, Imaging and Image Processing, ACTA Press, p. 480-484, 2002.

95. Ismail, M.A. and S. Gad, Off-line arabic signature recognition and verification. Pattern Recognition, volume 33, p. 1727-1740, 2000.

96. Renjini, L. and R., B., A Comprehensive Study on Handwritten Character Recognition System. IOSR Journal of Computer Engineering, volume 17, p. 01-07, 2015.

97. Azizi, N., Farah, N., Khadir, M. T., AND Sellama, M. Arabic handwritten word recognition using classifiers selection and features extraction/selection. In Proceedings of the 17th IEEE Conference in Intelligent Information System, p. 735-742, 2009.

98. Bay, H., T. Tuytelaars, and L. Van Gool, SURF: Speeded Up Robust Features, in Computer Vision – ECCV, Springer Berlin Heidelberg, p. 404-417, 2006.

99. Bay, H., et al., Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding, volume 110, p. 346-359, 2008.

100. Oyallon, E. and J. Rabin. Available from: http://www.ipol.im/pub/pre/69/preprint.pdf.

101. Lindeberg, T., Feature Detection with Automatic Scale Selection. International Journal of Computer Vision, volume 30, p. 79-116, 1998.

102. Madhvanath, S. and V. Govindaraju, The role of holistic paradigms in handwritten word recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 23, p. 149-164, 2001.

103. Madhvanath, S. and V. Govindaraju, Perceptual features for off-line handwritten word recognition: A framework for heuristic prediction, representation and matching, Advances in Pattern Recognition, Springer Berlin / Heidelberg. p. 524-531, 1998

104. Madhvanath, S. and V. Govindaraju. Serial classifier combination for handwritten word recognition. In Proceedings of the Third International Conference on Document Analysis and Recognition, p. 911-914, 1995.

105. Dehkordi, M.E., N. Sherkat, and R. Whitrow. Classification of off-line hand-written words into upper and lower cases. In Proceedings of the IEE Colloquium on Document Image Processing and Multimedia, p. 8/1 - 8/4, 1999.

106. Mori, S., K. Yamamoto, and M. Yasuda, Research on Machine Recognition of Handprinted Characters. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 6, p. 386-405, 1984.

107. Ho, T.K., J.J. Hull, and S.N. Srihari, A word shape analysis approach to lexicon based word recognition. Pattern Recognition Letters, volume 13, p. 821-826, 1992.

108. Farag, R.F.H., Word-Level Recognition of Cursive Script. IEEE Transactions on Computers, volume 28, p. 172-175, 1979.

109. Guillevic, D. and C.Y. Suen. Cursive script recognition applied to the processing of bank cheques. In Proceedings of the Third International Conference on Document Analysis and Recognition, p. 11-14, 1995.

110. Olivier, C., et al. Recognition of handwritten words using stochastic models. In Proceedings of the Third International Conference on Document Analysis and Recognition, p. 19-23, 1995.

111. Sharma, P.K. Multiple Classifiers for Unconstrained Offline Handwritten Numeral Recognition. In the International Conference on Computational Intelligence and Multimedia Applications, p. 344-348, 2007.

112. Steinherz, T., E. Rivlin, and N. Intrator, Offline cursive script word recognition – a survey. International Journal on Document Analysis and Recognition, volume 2, p. 90-110, 1999.

113. Madhvanath, S., G. Kim, and V. Govindaraju, Chaincode contour processing for handwritten word recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 21, p. 928-932, 1999.

114. Ho, T., A Theory of Multiple Classifier Systems and Its Application to Visual Word Recognition, PhD thesis, State University of New York, 1992.

115. Earnest, L.D., Machine Recognition of Cursive Writing. IFIP Congress, p. 462-466, 1962.

116. AlKhateeb, J.H., et al., Performance of hidden Markov model and dynamic Bayesian network classifiers on handwritten Arabic word recognition. Knowledge-Based Systems, volume 24, p. 680-688, 2011.

117. Al-Hajj Mohamad, R., L. Likforman-Sulem, and C. Mokbel, Combining Slanted-Frame Classifiers for Improved HMM-Based Arabic Handwriting Recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 31, p. 1165-1177, 2009.

118. Günter, S. and H. Bunke, HMM-based handwritten word recognition: on the optimization of the number of states, training iterations and Gaussian components. Pattern Recognition, volume 37, p. 2069-2079, 2004.

119. Mou-Yen, C., K. Amlan, and Z. Jian, Off-line handwritten word recognition using a hidden Markov model type stochastic network. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 16, p. 481-496, 1994.

120. Rodríguez-Serrano, J.A. and F. Perronnin, Handwritten word-spotting using hidden Markov models and universal vocabularies. Pattern Recognition, volume 42, p. 2106-2116, 2009.

121. Lowe, D., Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, volume 60, p. 91-110, 2004.

122. Rothacker, L., S.Vajda, and G.A.Fink. Bag-of-features representations for offline handwriting recognition applied to arabic script. In Proceedings of the 2012 International Conference on Frontiers in Handwriting Recognition (ICFHR '12), p. 149-154, 2012.

123. Pal, U., A. Belaid, and C. Choisy. Water reservoir based approach for touching numeral segmentation. In Proceedings of the Sixth International Conference on Document Analysis and Recognition, p. 892-896, 2001.

124. Chaudhuri, B., U. Pal, and M. Mitra, Automatic recognition of printed Oriya script. In Proceedings of the Sixth International Conference on Document Analysis and Recognition, volume 27, p. 23-34, 2002.

125. Pal, U. and S. Datta. Segmentation of Bangla unconstrained handwritten text. In Proceedings of the Seventh International Conference on Document Analysis and Recognition, p. 1128-1132, 2003.

126. Razzak, M.I., et al., HMM and fuzzy logic: A hybrid approach for online Urdu script-based languages' character recognition. Knowledge-Based Systems, volume 23, p. 914-923, 2010.

127. Dehkordi, M.E., N. Sherkat, and T. Allen, Handwriting style classification. International Journal on Document Analysis and Recognition, volume 6, p. 55-74, 2003.

128. Dolfing, J.G.A. A comparison of ligature and contextual models for hidden Markov model based on-line handwriting recognition. In Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, p. 1073-1076, 1998.

129. Cho, W., S.-W. Lee, and J.H. Kim, Modeling and recognition of cursive words with hidden Markov models. Pattern Recognition, volume 28, p. 1941-1953, 1995.

130. Milson, T.E. and K.R. Rao, A Statistical Model for Machine Print Recognition. IEEE Transactions on Systems, Man and Cybernetics, volume 6, p. 671-678, 1976.

131. Hamamura, T., T. Akagi, and B. Irie. An Analytic Word Recognition Algorithm Using a Posteriori Probability. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), p. 669-673, 2007.

132. Underwood, G. and K. Bargh, Word shape, orthographic regularity, and contextual interactions in a reading task. Cognition, volume 12, p. 197-209, 1982.

133. Wheeler, D.D., Processes in word recognition. Cognitive Psychology, Volume 1, p. 59-85, 1970.

134. Schlapbach, A. and H. Bunke. Using HMM based recognizers for writer identification and verification. In Proceedings of the Ninth International Workshop on Frontiers in Handwriting Recognition (IWFHR-9), p. 167-172, 2004.

135. Coetzer, J., B.M. Herbst, and J.A.d. Preez, Offline signature verification using the discrete radon transform and a hidden Markov model. Journal on Applied Signal Processing, p. 559-571, 2004.

136. Armand, S., M. Blumenstein, and V. Muthukkumarasamy. Off-line Signature Verification based on the Modified Direction Feature. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR 2006), p. 509-512, 2006.

137. Rabiner, L.R., A tutorial on hidden Markov models and selected applications in speech recognition In Proceedings of the IEEE, volume 77, p. 257-286, 1989.

138. Harmon, L.D., Automatic recognition of print and script. In Proceedings of the IEEE, volume 60, p. 1165-1176, 1972.

139. Pal, U. and B.B. Chaudhuri, Indian script character recognition: a survey. Pattern Recognition, volume 37, p. 1887-1899, 2004.

140. Fischer, A., et al., Lexicon-free handwritten word spotting using character HMMs. Pattern Recognition Letters, volume 33, p. 934-942, 2012.

141. Liang, H., et al. Keyword Spotting in Offline Chinese Handwritten Documents Using a Statistical Model. In Proceedings of the 2011 International Conference on Document Analysis and Recognition (ICDAR), p, 78-82, 2011.

142. Suwanwiwat, H. and M. Blumenstein, Short answer question examination using an automatic off-line handwritting recognition system and the Modified Direction Feature, In Proceedings of the 3rd International Conference on Machine Vision (ICMV 2010), p. 476-480, 2010.

143. Harlen, W. and M. James, Assessment and Learning: differences and relationships between formative and summative assessment. Assessment in Education: Principles, Policy & Practice, volume 4, p. 365 – 379, , 1997.

144. Mathias, H., The Evaluation of University Teaching: Context, Values and Innovation. Assessment & Evaluation in Higher Education, volume 9, p. 79 – 96, 1984.

145. Gay, L.R., The Comparative Effects of Multiple-Choice versus Short-Answer Tests on Retention. Journal of Educational Measurement, volume 17, p. 45-50, 1980.

146. Bacon, D.R., Assessing learning outcomes: A comparison of multiple-choice and short-answer questions in a marketing context. Journal of Marketing Education, volume 25, p. 31, 2003.

147. Scouller, K., The Influence of Assessment Method on Students' Learning Approaches: Multiple Choice Question Examination versus Assignment Essay. Higher Education, volume 35, p. 453-472, 1998.

148. J. Bull, C.M., Blueprint for computer-assisted assessment. London, 2004

149. S. Brown, P.R., J. Bull., Computer-assisted assessment in higher education. 1999, London: Kogan Page Limited.

150. DRS. 2011; Available from: http://www.drs.co.uk/.

151. Jackson, D., A semi-automated approach to online assessment, In Proceedings of the 5th annual SIGCSE/SIGCUE ITiCSE conference on Innovation and technology in computer science education, p. 164-167, Finland, 2000.

152. Rich, C.S. and W. Yihong. Online formative assessment using automated essay scoring technology in China and U.S.-- Two case studies. In Proceedings of the 2nd international Conference on Education Technology and Computer (ICETC '10), p. V3-524 - V3-528, 2010.

153. J. Burstein, C.L., R. Swartz, Automated Evaluation Of Essays And Short Answers. In Proceedings of the Fifth International CAA Conference, Loughborough, 2001.

154. Jain, A.K., R.P.W. Duin, and M. Jianchang, Statistical pattern recognition: a review. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 22, p. 4-37, 2000.

155. Liu, J. and P. Gader, Neural networks with enhanced outlier rejection ability for off-line handwritten word recognition. Pattern Recognition, volume 35, p. 2061-2071, 2002.

156. Maddouri, S.S. and H. Amiri. Combination of local and global vision modelling for Arabic handwritten words recognition. In Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition, p. 128-135, 2002.

157. Chaudhuri, B.B. and U. Pal, Skew angle detection of digitized Indian script documents. IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 19, p. 182-186, 1997.

158. Zanchettin, C., B.L.D. Bezerra, and W.W. Azevedo. A KNN-SVM hybrid model for cursive handwriting recognition. In Proceedings of the 2012 International Joint Conference on Neural Networks (IJCNN), p. 1-8, 2012.

159. Rabiner, L. and B. Juang, An introduction to hidden Markov models. ASSP Magazine, volume 3, p. 4-16, 1986.

160. Marti, U.V. and H. Bunke, Using a Statistical Language Model to Improve the Performance of an HMM-based Cursive Handwriting Recognition System. International Journal of Pattern Recognition & Artificial Intelligence, volume 15, p. 65, 2001.

161. Benouareth, A., A. Ennaji, and M. Sellami, Semi-continuous HMMs with explicit state duration for unconstrained Arabic word modeling and recognition. Pattern Recognition Letters, volume 29, p. 1742-1752, 2008.

162. Hai Rong, L., Y. Wen Jun, and D. Jin. Off-line signature verification based on deformable grid partition and Hidden Markov Models. In Proceedings of the IEEE International Conference on Multimedia and Expo (ICME 2009), p. 374-377, 2009.

163. Frinken, V., et al., Improved Handwriting Recognition by Combining Two Forms of Hidden Markov Models and a Recurrent Neural Network, Computer Analysis of Images and Patterns, p. 189-196, 2009.

164. Plötz, T. and G. Fink, Markov models for offline handwriting recognition: a survey. International Journal on Document Analysis and Recognition, volume 12, p. 269-298, 2009.

165. Nag, R., K. Wong, and F. Fallside. Script recognition using hidden Markov models. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '86), p. 2071-2074, 1986.

166. El-Yacoubi, A., et al., An HMM-based approach for off-line unconstrained handwritten word modeling and recognition. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 21, p. 752-760, 1999.

167. Chen, F.R., L.D. Wilcox, and D.S. Bloomberg. A comparison of discrete and continuous hidden Markov models for phrase spotting in text images. In Proceedings of the Third International Conference on Document Analysis and Recognition, p. 398-402, 1995.

168. Ahmad, I., Rothacker, L., Fink, G. A., Mahmoud, S. A., Novel Sub-character HMM Models for Arabic Text Recognition. In Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR ‘13), p. 658-662, 2013.

169. Lorigo, L.M. and V. Govindaraju, Offline Arabic handwriting recognition: a survey. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 28, p. 712-724, 2006.

170. Huang, X.D. and M.A. Jack, Semi-continuous hidden Markov models for speech signals. Computer Speech & Language, volume 3, p. 239-251, 1989.

171. Espana-Boquera, S., et al., Improving Offline Handwritten Text Recognition with Hybrid HMM/ANN Models. In Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 1-1, 2010.

172. Justino, E.J.R., F. Bortolozzi, and R. Sabourin, A comparison of SVM and HMM classifiers in the off-line signature verification. Pattern Recognition Letters, volume 26, p. 1377-1385, 2005.

173. Zhu, B., et al., A robust model for on-line handwritten Japanese text recognition. In Proceedings of the International Journal on Document Analysis and Recognition (IJDAR), volume 13, p. 121-131, 2010.

174. Milgram, J., M. Cheriet, and R. Sabourin. “One Against One” or “One Against All”: Which One is Better for Handwriting Recognition with SVMs? In Proceedings of the Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule (France), 2006.

175. Nemmour, H. and Y. Chibani. Handwritten Arabic word recognition based on Ridgelet transform and support vector machines. In Proceedings of the 2011 International Conference on High Performance Computing and Simulation (HPCS), p. 357-361, 2011.

176. Bahlmann, C., B. Haasdonk, and H. Burkhardt. Online handwriting recognition with support vector machines - a kernel approach. In Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition, p. 49-54 , 2002.

177. Papavassiliou, V., et al., Handwritten document image segmentation into text lines and words. Pattern Recognition, volume 43, p. 369-377, 2010.

178. Al-Boeridi, O., S.M. Syed Ahmad, and S.P. Koh, A scalable hybrid decision system (HDS) for Roman word recognition using ANN SVM: study case on Malay word recognition. Neural Computing and Applications, p. 1-9, 2015.

179. Rajashekararadhya, S.V. and P.V. Ranjan. Support Vector Machine based Handwritten Numeral Recognition of Kannada Script. In Proceedings of the 2009 IEEE International in Advance Computing Conference (IACC 2009), p. 381-386, 2009.

180. Impedovo, D. and G. Pirlo, Automatic Signature Verification: The State of the Art. In Proceedings of the IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, volume 38, p. 609-635, 2008.

181. Chen, M.Y., A. Kundu, and J. Zhou. Off-line handwritten word recognition (HWR) using a single contextual hidden Markov model. Proceedings of the 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p. 669-672, 1992.

182. Suwanwiwat, H., et al. Off-line handwritten Thai name recognition for student identification in an automated assessment system. Proceedings of the 2014 International Joint Conference on Neural Networks (IJCNN), p. 2347-2353, 2014.

183. Suwanwiwat, H., et al. Off-Line Handwritten Bilingual Name Recognition for Student Identification in an Automated Assessment System. Proceedings of the 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), p. 271-276, 2014.

184. Armand, S., M. Blumenstein, and V. Muthukkumarasamy, Off-line Signature Verification Using an Enhanced Modified Direction Feature with Single and Multi-classifier Approaches. Computational Intelligence Magazine, volume 2, p. 18-25, 2007.

185. Blumenstein, M., X.Y. Liu, and B. Verma. A modified direction feature for cursive character recognition. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks, volume 4, pages 2983-2987, 2004.

186. Liu, C.-L. and H. Fujisawa, Classification and Learning Methods for Character Recognition: Advances and Remaining Problems, in Machine Learning in Document Analysis and Recognition, p. 139-161, 2008.

187. Riedmiller, M. and H. Braun. A direct adaptive method for faster backpropagation learning: the RPROP algorithm. In Proceedings of the IEEE International Conference on Neural Networks, volume 1, p. 586-591, 1993.

188. Chen, C.-s. and S.-l. Su. Resilient Back-propagation Neural Network for Approximation 2-D GDOP. In Proceedings of the Multiconference of Engineerings and Computer Scientists, Hong Kong, 2010.

189. Chih-Wei, H. and L. Chih-Jen, A comparison of methods for multiclass support vector machines., In proceedings of the IEEE Transactions on Neural Networks, volume 13, p. 415-425, 2002.

190. Yoonkyung Lee, Yi Lin, and G. Wahba, Multicategory Support Vector Machines, theory, and application to the classification of microarray data and satellite radiance data. Journal of the American Statistical Association, volume 99, p. 67-81, 2004.

191. Crammer, K. and Y. Singer, On the algorithmic implementation of multiclass kernel-based vector machines. The Journal of Machine Learning Research, volume 2, p. 265-292, 2002.

192. Schölkopf, B., K. Tsuda, and J.P. Vert,. Kernel Methods in Computational Biology. Bradford Bks, 2004.

193. Hall, M., et al., The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter, volume 11, p. 10-18, 2009.

194. Chang, C.-C. and C.-J. Lin, LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology (TIST), volume 2, p. 1-27, 2011.

195. Nopsuwanchai, R., Discriminative training methods and their applications to handwriting recognition, Technical Report, University of Cambridge, computer laboratory. 2005

196. Kennard, D.J., W.A. Barrett, and T.W. Sederberg. Word Warping for Offline Handwriting Recognition. In Proceedings of the 2011 International Conference on Document Analysis and Recognition (ICDAR), p. 1349-1353, 2011.

197. Ruiz-Pinales, J., R. Jaime-Rivas, and M.J. Castro, Discriminative Capacity of Perceptual Features in Handwriting Recognition. Telecommunications and Radio Engineering, volume 64, p. 931-937, 2005.

198. Patel, M.S., S. Reddy, and A. Naik, An Efficient Way of Handwritten English Word Recognition, In Proceedings of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA 2014), p. 563-571, 2015.