A Writer Identification System for Handwritten Malayalam Documents

Thesis submitted by

SREERAJ M

In partial fulfilment of the
requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

UNDER THE FACULTY OF TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE
COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY
Cochin - 682 022
INDIA

July 2012

A Writer Identification System for Handwritten Malayalam Documents

Ph.D Thesis in the field of Pattern Recognition

Author:
Sreeraj M
Department of Computer Science
Cochin University of Science and Technology
Cochin - 682 022, Kerala, India
sreerajtkzy@gmail.com

Supervisor:
Dr. Sumam Mary Idicula
Professor
Department of Computer Science
Cochin University of Science and Technology
Cochin - 682 022, Kerala, India
sumam@cusat.ac.in

July 2012

CERTIFICATE

Certified that the work presented in this thesis entitled **"A Writer Identification System for Handwritten Malayalam Documents"** is a bonafide work done by Mr. Sreeraj M, under my guidance in the Department of Computer Science, Cochin University of Science and Technology and that this work has not been included in any other thesis submitted previously for the award of any degree.

Kochi
July 30, 2012

Dr. Sumam Mary Idicula
(Supervising Guide)

DECLARATION

I hereby declare that the work presented in this thesis entitled **"A Writer Identification System for Handwritten Malayalam Documents"** is based on the original work done by me under the guidance of Dr. Sumam Mary Idicula, Professor, Department of Computer Science, Cochin University of Science and Technology and has not been included in any other thesis submitted previously for the award of any degree.

Kochi
July 30, 2012 Sreeraj M.

Acknowledgements

The research work leading to PhD is a very intense process, where many times one feels lost or needs to take critical decisions. On those moments it is great to have the support and words of wisdom from people who can very quickly understand the problems one facing. I would like to thank the people that were with me during the last four years and those who helped me to perform my work.

I start by thanking my guide Dr. Sumam Mary Idicula, Professor, Department of Computer Science, Cochin University of Science and Technology, for always providing me very constructive comments and suggestions. I am privileged to have her as my guide. Her rich experience and good intentions are sparkles in the thesis. It was a pleasure to work with her, and I certainly learned a lot with her.

Let me thank Dr. Poulose Jacob, Professor, Head of Department, Department of Computer Science, Cochin University of Science and Technology, for his support in me pursuing the PhD programme in the department. I am grateful for his suggestions in my learning process.

I would like to extent my sincere gratitude to Dr.G. Santhosh Kumar, Assistant Professor, Department of Computer Science, Cochin University of Science and Technology, from whom I have learnt quite a few things. Apart from the interesting research discussions that we had, it was a pleasure for me to discuss with him many other issues.

Throughout this journey I have had the privilege of interacting with two other very special people with the help of my uncle Mr. Sreesakumar. One was Dr. Vishnu potty, former Director of Forensics, from whom I have got valuable feedback in handwriting analysis which turned to be turning points in my research work. I express my sincere gratitude to him.

Knowledge of an outdated script is uncommon. When I faced with the problem of understanding Grantha script I struggled a lot to find a person who knew it. Mr.V. Manmadhan Nair, former Director, Department of Archeology probably the only living expert in the Grantha script helped me in this regard.I express my heartfelt thanks to him for spending time to teach me the Grantha Script.

I acknowledge Prof. K.V. Pramod, Prof. B. Kannan and Prof. A. Sreekumar of the Department of Computer Applications for always giving me zero barrier access for discussions whenever I faced problems.

I had the support and guidance of Mr. Joseph.V.Mathews and Ms.Glory Thomas from whom I got constant inspiration to work hard and to be persistent. They always kept me motivated and inspired during this entire journey.

I would like to thank Ms. Saritha.S, Mr. Binu. A and Mr. Sreekumar.K who have contributed immensely to the ideas, concepts and prototypes described this thesis. I would also want to thank Ms. Sariga Raj and Vinu Paul. M for their constructive comments on my thesis.

A special gratitude and appreciation is extended to all the enthusiastic students of those Schools in Alappuzha district who had contributed the handwritten samples for building the database for my research work. Also my friends at the Department of Computer Science had contributed to it. I am deeply indebted to them for this.

I specially thank Mr. Joe Joseph for providing very good library support. I also thank technical staffs, Mr. Renjith, Mr. Shibu, & Mrs. Manju for providing me all the technical support required for carrying out my research work. I am grateful to all the staff of the department for their encouragement and support.

I thank my friends for the support they had given me in difficult moments. Unfortunately, some of them are far away but the current communication technologies made our contacts easier.

I also want to share this special moment with my parents K. Madhavan kutty and T.S. Sreedevi who kept showing me that unconditional support, pushing me to pursue my dreams every morning, ignoring the fact that I was not there when they needed me. I could never have reached this point without their love and nourishment.

Sreeraj M.

Abstract

Handwriting is an acquired tool used for communication of one's observations or feelings. Factors that influence a person's handwriting not only dependent on the individual's bio-mechanical constraints, handwriting education received, writing instrument, type of paper, background, but also factors like stress, motivation and the purpose of the handwriting. Despite the high variation in a person's handwriting, recent results from different writer identification studies have shown that it possesses sufficient individual traits to be used as an identification method.

Handwriting as a behavioral biometric has had the interest of researchers for a long time. But recently it has been enjoying new interest due to an increased need and effort to deal with problems ranging from white-collar crime to terrorist threats. The identification of the writer based on a piece of handwriting is a challenging task for pattern recognition. The main objective of this thesis is to develop a text independent writer identification system for Malayalam Handwriting. The study also extends to developing a framework for online character recognition of Grantha script and Malayalam characters.

The writer identification system proposed in this thesis comprises of different phases like image preprocessing, feature extraction, training and classification or identification. The feature extraction phase includes three schemes. One is at the grapheme level, next Character level and third at the document level. The performance of the overall system is measured using statistical measurements. In order to analyse the system performance, experiments are carried out with different classifiers like Naive Byes, KNN, SVM and Adaboost. The comparison of results are based on the identification rate of classifiers, stability of features and classifiers, consistency measurements,

influence of single features, and cumulative features. From each of these schemes elegant /decisive features that distinguish a writer were obtained.

A system that can recognize online handwritten Malayalam characters utilizing the optimum decisive features obtained from above schemes was developed. Further comparisons were made with the system developed and systems using other features and classifiers. Results showed that the system developed with the decisive features performed better.

The analysis and recognition of historical documents have attracted interest recent years. This may be due to digitization drive for preservation of documents that embody the artistic, cultural and technical heritage of a country. With this in mind the thesis proposes a system to recognize handwritten Grantha script and obtain its Malayalam equivalent. This has significance because the root script of Malayalam is the Grantha script. The system adopts the same framework as that of Online Malayalam Character Recognition.

Contents

List of Figures

List of Tables

List of Symbols

χ^2	Chi-square distance
Δt	Small variation in time
κ	Curvature at a point
σ	Standard deviation
μ	Arithmetic mean
C_v	Coefficient of variation
${Fv_i}^w$	i^{th} feature vector of a writer w
v_x, v_y	Velocity of x and y direction
$\hat{x}_i^{'}, \hat{y}_i^{'}$	First order derivative of (x_i, y_i)
$\hat{x}_i^{''}$, $\hat{y}_i^{''}$	Second order derivative of (x_i, y_i)
$\prod_{f_i \in D} p\left(f_i / W_i\right)$	joint probability density function (with respect to a product measure) as a product of the individual density functions(each feature), conditional on their parent variable(writer)
$P\left(W_i / Q_t\right)$	Conditional probability of i^{th} writer(the probability of i^{th} writer, given test query document Q_t)
$Sim\left(Fv_i^w, Fv_t^Q\right)$	Similarity function between two feature vectors
$LBP_{s,\Psi}^{p,r}\left(m,n\right)$	The LBP code of each wavelet sub band
$G(x, y, \alpha)$	Variable scale Gaussian function

List of Abbreviations

dB4	Daubechies wavelets
WD-LBP	Wavelet Domain Local Binary Patterns
SIFT	Scale Invariant Features Transform
k-NN	K Nearest Neighbor classifier
SVM	Support Vector Machine
SOM	Self-Organizing Maps
PDF	Probability Density Function
RBF	Radial Basis Function
DTW	Dynamic Time Warping
JNI	Java Native Interface
df	degrees of freedom in regression analysis
SS	Sum of the Squares in regression analysis
MS	Mean Square in regression analysis
F	F test (ratio of the mean squares) in regression analysis

Chapter 1

Introduction

1.1 Overview

Writing is defined as "the representation of language in a textual medium through the use of a set of signs or symbols". History describes writing as a consequence of political expansion in ancient cultures, which needed reliable means for transmitting information, maintaining financial accounts, keeping historical records, and similar activities. Any language has its history of evolution and development. Languages undergo changes time to time and the recorded thoughts or knowledge (written form of a language) can be an unknown sea if the language becomes extinct or not in use. That is, initially a language is an expression of

thoughts by sound, means a spoken language. On the invention of scripts, written language has been developed, and the evolution goes on. And a distinguishing character is left with each period, rare or tribe. Recognizing or identifying a language of a particular period or of a particular ethnic group has further developed, as language and its purposes grown, to recognize/ identify the writers by the distinctive characteristics of them [1].

Each person has his own manner of writing which depends on a lot of factors like specific shape of letters, spacing between letters, slope, pressure to the paper, average size of letters and so on. Handwriting of a person is also dependent on the mental state of the person like his level of motivation, anger, happiness and others. But it is found that handwriting of a person is relatively stable though may be affected slowly with age.

This uniqueness in handwriting style is exploited in addressing concerns about potential authorship of "questioned documents". Recently many crimes have clues in certain inscriptions or handwritten notes. Deciphering the authorship could prove to be the vital turning point in solving/ averting danger of such cases. The forensics department considers this branch of study as most challenging one and many promising research has been done all over the world owing to the fact that there are thousands of scripts in the world.

Most studies about writer identification are based on the documents in English/ Anglo Saxon, Chinese, Arabic, Persian or related languages. With the distinctive characteristics of Indian languages, the tasks on character recognition and writer identification are yet to be developed. And for the ethnic Dravidian-South Indian- languages, it is in its infancy stage. Malayalam, a unique Dravidian language reserves its own identity rooted to Grantha-Brahmi scripts. Identifying decisive features for descriptive

analysis of writings in Malayalam is a novel approach.

Writer identification, in general is important in forensic and related branches of science, digital rights administration, forensic expert decision-making systems, document analysis methods for authentication systems and writer verification schemes. The parameters generally considered are universality uniqueness, aging, availability, processing complexity and acceptability. The same is applicable in Malayalam too with an additional significance of finding the evolution and development of Malayalam language, proving its strong relationship with Grantha script and introducing a common, unique system to derive both the languages (Malayalam & Grantha). The system designed and composed of different phases like image preprocessing, feature extraction, training and classification or recognition.

1.2 Characteristics of Malayalam script

Malayalam is a Dravidian language spoken by about 35 million people. It is spoken mainly in the state of Kerala and in the Lakshadweep Islands. Malayalam is originated from proto Dravidian in the 6^{th} century. Although Malayalam is a Dravidian language, during the ages it has been mainly Sanskritised and now, over 80% words of modern Malayalam are from pure Sanskrit. Malayalam first appeared in writing in the *vazhappalli* inscription (830 A.D). Later it has been developed into *vattezhuth*. When the sanskritation in effect, it was the advent of Grantha script and the Grantha- Malayalam was *aarya-ezhuthu*. Malayalam became an independent language from 9^{th} century A.D.

Malayalam script has the following features. It has syllabic alphabet in which all consonants have an inherent vowel. Diacritics can appear above, below, before or after the consonant they belong to and are used

to change the inherent vowel. When certain consonants occur together, special conjunct symbols are used which combine the essential parts of each letter. There are about 128 characters in the Malayalam character set which includes vowels (15), consonants (36), chillu (5), anuswaram, visargam, chandrakkala-(total-3), consonant signs (3), left vowel signs (2), right vowel signs (7), conjunct consonants (57). Out of all these characters mentioned, only 64 of them are considered to be the basic ones which is shown in Fig.1.1 [2].

Vowels

അ	ആ	ഇ	ഉ	ഋ	എ	ഏ	ഒ

Consonants

ക	ഖ	ഗ	ഘ	ങ	
ച	ഛ	ജ	ഝ	ഞ	
ട	ഠ	ഡ	ഢ	ണ	
ത	ഥ	ദ	ധ	ന	
പ	ഫ	ബ	ഭ	മ	
യ	ര	ല	വ	ശ	
ഷ	സ	ഹ	ള	ഴ	റ

Dependent Vowel Signs

ാ	ി	ീ	ു	ൂ	ൃ	െ	േ	ൗ

Anuswaram	Visargam	Chandrakala
ം	ഃ	്

Consonant Signs

ൈ	ൊ	ോ

Chillu

ൻ	ൽ	ർ	ൾ	ൺ

Figure 1.1: 64 basic characters of Malayalam

The properties of Malayalam characters are the following

- Since Malayalam script is an alphasyllabary of the Brahmic family they are written from left to right.

- Almost all the characters are circular by themselves. They consist of loops and curves. The loops are written frequently in the clockwise order.

- Several characters are different only by the presence of curves and loops.

- Unlike English, Malayalam scripts are not case sensitive and there is no cursive form of writing.

- Malayalam is a language which is enriched with vowels, consonants and has the maximum number of sounds that are not available in many other languages as shown in Fig.1.2.

ണ, ള, ഴ, റ, ഓ, ഏ, ശ, ൺ, ർ

Figure 1.2: Rare Sounds of scripts available only in Malayalam language.

- Two prominent ways of writing Malayalam scripts exists today. One followed by older generation and the other followed by younger generation. But the latter has become standard form even though usage of the former is still common. Some samples are given in Fig.1.3.

Old scripts	New scripts
കൃ	കൂ
ഗ്ല	ഗ്ള
ത്ഥ	ത്ഥ
ച്ച	ച്ച
ന്ത	ന്ത

Figure 1.3: Example of Old and new scripts of Malayalam.

1.3 Challenges in Malayalam script

Motivation for a Malayalam writer identification scheme stems out from the following challenges posed by the language as well as some other factors.

i. **Meager allographic variation of writers in Malayalam documents:** A major factor causing variation in handwriting is allographic variation. Writer specific character shapes are derived from this variation. They are a threat to automatic script recognition. In spite of this, it substitutes vital information for writer identification. Due to the curvaceous nature of Malayalam characters this variation is very low in Malayalam handwriting. Same sentences written by three writers given in Fig. 1.4 can impart a feel of this argument.

ii. **Insufficient discriminating capacity of a single character in Malayalam language:** A single character does not provide sufficient discriminating values and hence a combination of characters may be necessary to give out a prominent feature vector of the handwriting. This again adds to identification complexity.

iii. **Non-existence of uppercase and lowercase in Malayalam language:** Writers adopt certain prominent styles related to upper case and lower case characters. Since Malayalam scripts do not have upper and lower cases this prominent discrimination cannot be applied. Same is the case with cursive style and Malayalam has no cursive form of writing.

iv. **Absence of dataset:** Absence of the dataset of handwritten pages of different users in Malayalam pose a great challenge. Hence, a collection of handwritings of different users of similar as well as different data had to be collected for the purpose of implementation.

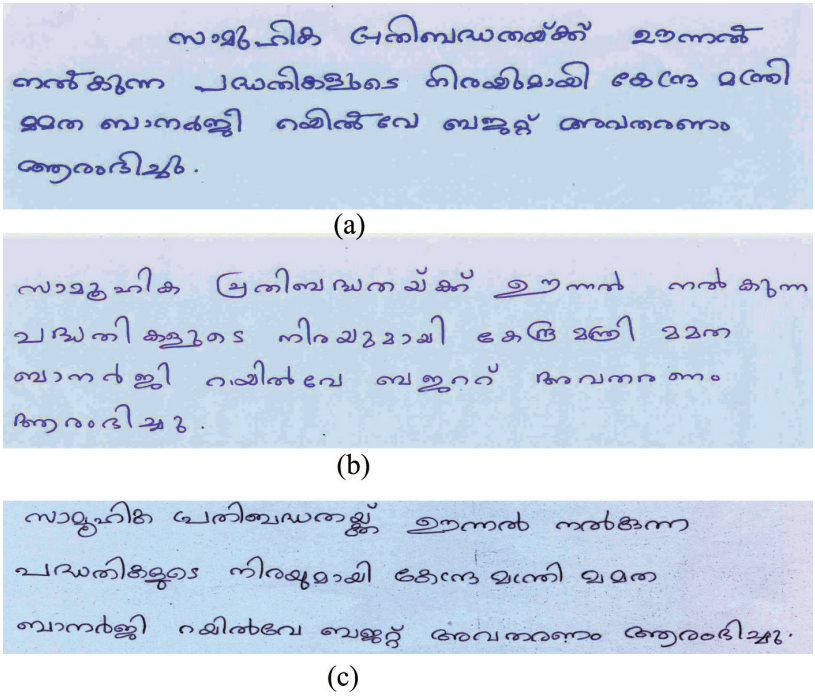


Figure 1.4: Allographic variation of three writers in Malayalam: (a) Extracted from Writer 1 (b) Extracted from Writer 2 (c) Extracted from Writer 3

v. **Writing impression:** Pen grip, the orientation of the wrist and the fingers together constitute a habitual parameter slant (shear) in the writing style of each user [3]. Malayalam script mainly contains loops and curves wherein every variation has to be considered. When different writings are compared this parameter is low. So there is a need of observing the minute changes in affine transform in the loops and curves of each character in Malayalam script.

1.4 Problem Statement

The shape and style of writing varies from one person to another. Even for one particular person, his or her writing's shape and style can be different at

different times. At present writer identification of handwritten Malayalam documents is done manually. Malayalam handwritten characters have to be analyzed to obtain decisive features to identify individual handwriting style. The challenges posed on the Malayalam characters made it difficult to acquire minimum variation in intra-class and maximum variation in inter-class with reference feature vectors. Thus, the main research question is to identify features of Malayalam handwriting, which can preserve the individuality of handwriting in data representation. Tackling this challenging problem raises a number of important sub research questions like:

i. How can individual handwriting style be characterized using computer algorithms?

ii. What representations or features are most appropriate or elegant for Malayalam script and how can they be combined effectively?

iii. How much efficiency can be achieved by automatic methods of writer identification?

iv. Can the feature identified for writer identification be used for developing a framework for further applications like online handwritten character recognition, historical document analysis etc.

1.5 Objectives and Scope

Identifying the writer of a handwritten document using automatic image-based methods is an interesting pattern recognition problem with direct applicability in the field of forensic and historic document analysis. To achieve this, the objectives of this research work are identified as follows.

i. Identify the features that can represent the writer individuality characteristics at three levels such as grapheme, character and document level.

ii. Compare the impact of various features identified for Malayalam language at the three levels mentioned above.

iii. Obtain an elegant feature set for Malayalam handwritten characters by analyzing the result of writer identification system at three levels.

iv. Develop a framework based on the elegant feature derived above to build further applications like online recognition system of handwritten Malayalam characters and ancient Grantha script.

v. Evaluate the performance of the framework by building various classification models developed.

1.6 Contribution of the Thesis

The contributions of this thesis can be summarized as follows.

i. A writer identification scheme for Malayalam language was designed, developed, implemented and tested. The system was designed and comprised of different phases like image preprocessing, feature extraction, training and classification or recognition. This was done in three levels, *viz.*, grapheme, character and document level. This system formed a basis for writer identification in Malayalam, as well as in its root script , Grantha.

ii. In the process of writer identification, the elegant/decisive features to represent distinctly each writer at all levels like grapheme, character and document have been identified and the impact of those elegant features are outlined in terms of experiments conducted on data sets.

The experiments were carried out with relevant functional points from the perspective of recognition rate of classifiers, amount of data set, similarity measurements, single feature, and cumulative features and so on.

iii. A system for the online recognition of handwritten Malayalam characters and Grantha scripts is evolved from the basic writer identification scheme designed and implemented.

1.7 Outline of the Thesis

The following eight chapters in Fig.1.5 compile the structure of the thesis.

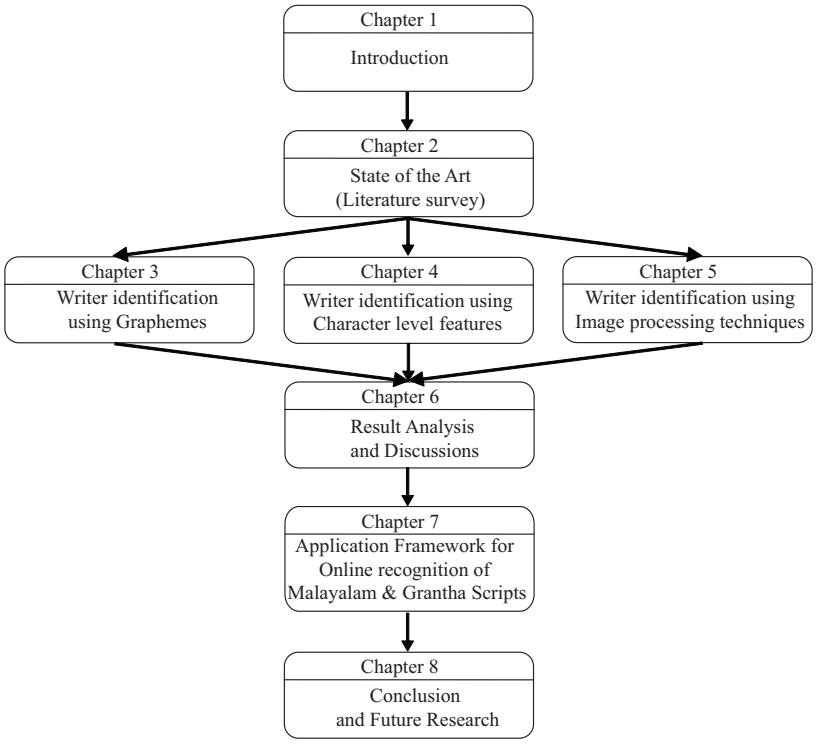


Figure 1.5: Structure of thesis

Chapter 2

A Survey on Writer Identification Schemes

This chapter presents a survey of the literature on writer identification schemes and techniques available till date. The content here outlines an overview of the writer identification schemes mainly in Chinese, English, Arabic and Persian languages. Taxonomy of different features adopted for online and offline writer identification schemes are also drawn. The feature extraction methods adopted for the schemes are discussed in length outlining the merits and demerits of the same. In automated writer identification, text independent and text dependent methods are available which is also discussed here. An evaluation of writer identification schemes under multiple languages is also analyzed by comparing the identification rate.

2.1 Introduction

The growth of artificial intelligence and pattern recognition fields owes greatly to one of the highly challenged problem of handwriting identification. Identifying the handwriting of a writer is highly essential today due to the immense growth in technology and its applications in wide areas. The application of writer identification can be seen in wide areas, such as, digital rights management in the financial sphere, forensic expert decision-making systems etc. By combining identification with writer verification an authentication system can be developed which can be used to monitor and regulate the access to certain confidential sites or data where large amounts of documents, forms, notes and meeting minutes are constantly being processed and managed. The knowledge of the identity of the writer would provide an additional value to the system. It can also be used for historical document analysis [4], handwriting recognition system enhancement [5] and hand held and mobile devices [6]. To a certain extent its recent development and performance are considered as a strong tool for physiologic modalities of identification, such as DNA and fingerprints [7].

It is evident that the importance of writer identification has become more significant these days. Obviously, the number of researchers involved in this challenging problem is going high as a result of these opportunities. There are numerous languages throughout the world. Each language poses a different threat to the writer identification problem depending on the characteristics of the language. So it is very clear that the identification problem varies across multiple languages.

The handwriting-based writer identification is an active research arena. As it is one of the most difficult problems encountered in the field of computer vision and pattern recognition, the handwriting-based writer identification problem faces with a number of sub problems like

(i) Designing algorithms to identify handwritings of different individuals

(ii) Identifying relevant features of the handwriting

(iii) Basic methods for representing the features

(iv) Identifying complex features from the basic features developed

(v) Evaluating the performance of automatic methods

The rest of the chapter is organized as follows. The state of the art in writer identification in languages like Chinese, English, Arabic and Persian is presented in detail. Also a taxonomy for online and offline writer identification depending on features is depicted. The performance evaluation of various writer identification schemes across multiple languages also is tabulated.

2.2 Writer identification - The State of the Art

A comprehensive review of automatic writer identification till 1989 is given in [8]. As an extension, the work from 1989 -1993 is published in [9]. Fig.2.1 describes the standard framework of writer identification [10]. The necessary features from the handwritten documents are extracted as the first step. Subsequently the features extracted are used to identify the writer of the document using similarity score method. The writer with high similarity score is considered as the writer of the document.

Based on the method of writing, automated writer identification has classified into on-line and off-line. The on-line writer identification task is considered to be less difficult than the offline one as it contains more information about the writing style of a person, such as speed, angle and pressure, which are not available in the off-line one [7][11]. Based on

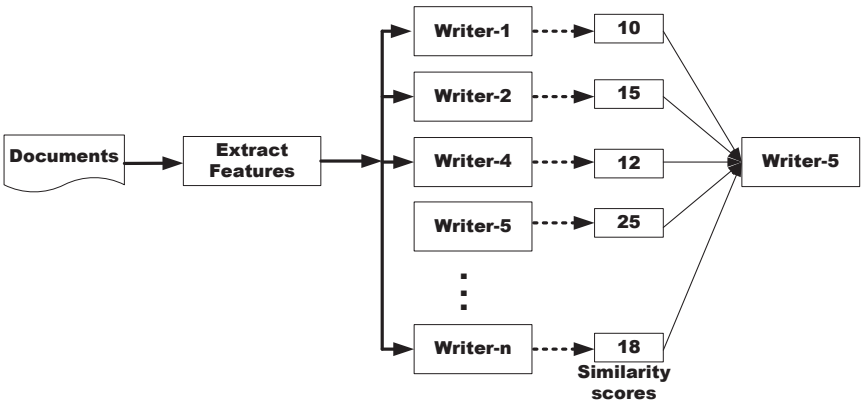


Figure 2.1: Writer Identification framework

the different features associated with the writing, a taxonomy has been developed and it is given in Fig.2.2.

Text-dependent & text-independent is another type of classification for automated writer identification. Dependending on the text content, text-dependent methods matches the same characters and hence requires the writer to write the same text. The text-independent methods are able to identify writers independent of the text content and it does not require comparison of the same characters. Thus it is very similar to signature verification techniques and uses the comparison between individual characters or words of known semantic (ASCII) content. This method considers the global style of hand writing the metric for comparison, and produces better identification results. Since text-dependent method requires the same writing content this method is not apt for many practical situations. Even though it has got a wider applicability, text-independent methods do not obtain the same high accuracy as text-dependent methods do.The following section describes the various approaches used for writer identification in different languages.

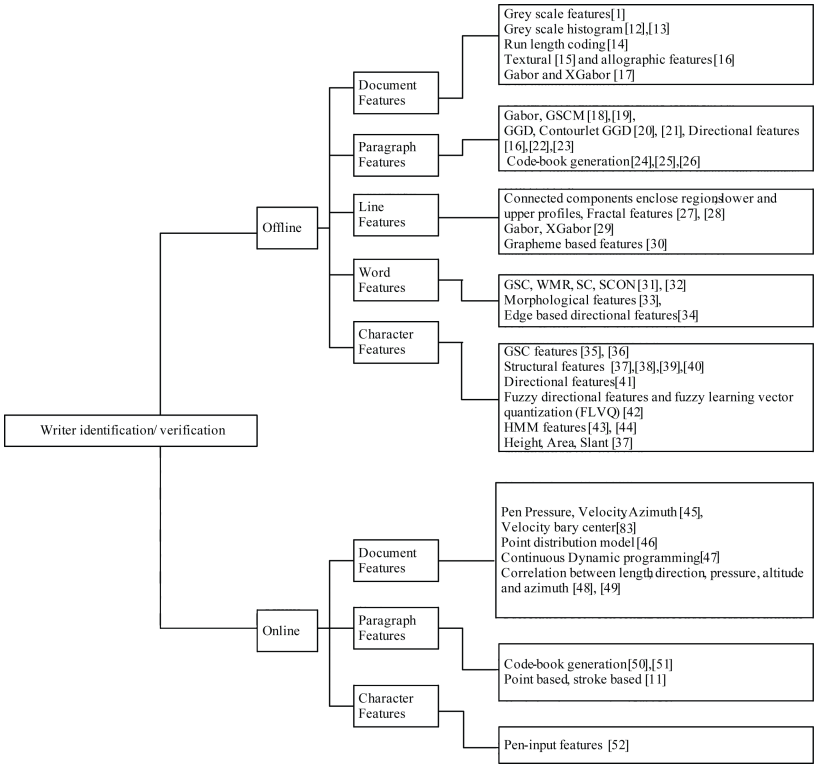


Figure 2.2: Taxonomy of Writer Identification

2.3 Chinese, English and other languages

In the end of nineties, Said et al. [19] [53] proposed a text-independent approach for writer identification that derives writer-specific texture features using multichannel Gabor filtering and Gray-Scale Co-occurrence Matrices. The framework required uniform blocks of text that are generated by word deskewing, and also setting a predefined distance between text lines/words and text padding. Two sets of twenty writers and 25 samples per writer were used in the experiment. Nearest centroid classification using weighted Euclidean distance and Gabor features achieved 96% writer identification accuracy, thus revealing that the two-dimensional Gabor model outperformed gray-scale co-occurrence matrix. A similar approach has also been used on machine print documents for script [54] and font [55] identification.

Zois and Anastassopoulos [33] implemented writer identification in 2000 and verified using single words. Experiments were performed on a data set of 50 writers. The word "characteristic" was written 45 times by each writer, both in English and in Greek. After image thresholding and curve thinning, the horizontal projection profiles were resampled, divided into 10 segments, and processed using morphological operators at two scales to obtain 20-dimensional feature vectors. Classification was performed using either a Bayesian classifier or a multilayer perceptron. The system showed an accuracy of 95% for both English and Greek words. In the writer identification scheme suggested by Marti et al. [28] and Hertel and Bunke [27], text lines were the basic input unit from which text-independent features were computed using the height of the three main writing zones, slant and character width, the distances between connected components, the blobs enclosed inside ink loops, the upper/lower contours, and the thinned trace processed using dilation operations. Using

a k-nearest-neighbour classifier, identification rates exceeded 92% in test cases on a subset of the IAM database [56] with fifty writers and five handwritten pages per writer.

Graham Leeham et al. proposed a methodology to identify the writer of numerals [37]. The features included parameters such as height, width, area, center of gravity, slant, number of loops, etc. The system was tested among fifteen people and the accuracy was 95%. However to determine the precise accuracy it should be verified across larger databases. Srihari et al. [1], [57] proposed a large number of features for the writing which can be classified into two categories. a) Macrofeatures - They operate at document/paragraph/word level. The parameters used are gray-level entropy and threshold, number of ink pixels, number of interior/exterior contours, number of four-direction slope components, average height/slant, paragraph aspect ratio and indentation, word length, and upper/lower zone ratio. b) Microfeatures - They operate at word/character level. The parameters comprise of gradient, structural, and concavity (GSC) attributes. These features were used originally for handwritten digit recognition [58]. Text-dependent statistical evaluations were performed on a data set containing thousand writers who copied a fixed text of 156 words (the CEDAR letter) three times. This is the largest data set ever used till now in writer identification methodologies. Microfeatures outperform macrofeatures in identification tests with an accuracy exceeding 80%. A multilayer perceptron or parametric distributions were used for writer verification with an accuracy of about 96%. Writer discrimination was also done using individual characters [35], [36] and using words [31], [32].

Bensefia et al. [24], [59], [60], [61] use graphemes generated by a handwriting segmentation method to encode the individual characteristics of handwriting independent of the text content. Grapheme clustering was

used to define a feature space common for all documents in the data set. Experimentations were done on three data sets containing 88 writers, 39 writers (historical documents), and 150 writers, with two samples (text blocks) per writer. Writer identification was performed in an information retrieval framework, while writer verification was based on the mutual information between the grapheme distributions in the two handwritings which were used for comparison. Concatenations of graphemes are also analyzed in the mentioned papers. An accuracy of about 90% was reported on the different test data sets. A feature selection study is also performed in [62].

In [24], [59] Ameur Bensefia et al. have developed a probability based approach using a codebook of graphemes in the IAM and PSI databases. The system accuracy was 95% in IAM database and 86% in PSI database. Also, Laurens van der Maaten et al. have used a combination of single directional features and codebook of graphemes [63]. The method was tested on 150 writers and the system accuracy was 97%. Vladimir Pervouchine *et al.* only focused on letters "t" and "h" on their English identification system. After detecting these shapes in the image, their skeletons were extracted. A cost function along the curve is then calculated and the similarity of cost functions identifies the writer [62]. It is obvious that this method cannot be extended for other languages. Schomaker et al. has presented a method based on fragmented connected-component contours (FCO3) [65], [66]. They used the method in the classification phase to calculate distance. Also, they tested it in an English data set with 150 writers. The top-1 of the method results had 72% and the top-10 had 93% accuracy. However, the top-10 results were satisfactory but its top-1 is not.

Schlapbach et al. implemented an HMM based writer identification and verification method [43], [44]. An individual HMM was designed

and trained for each writer's handwriting. To determine which writer has written an unknown text, the text is given to all the HMMs. The one with biggest result is assumed to be the writer. The identification method was tested by using documents gathered from 650 writers. The identification accuracy was 97%. Also, this method was tested as a writer verification method. This was achieved by a collection of writings from 100 people and twenty unskilled and twenty skilled imposters, who forged the originals. Experimental results obtained showed about 96% overall accuracy in verification. Thus it is obvious, that this method can be extended to other languages by applying some changes in feature extraction phase. The difference between the two writer identifications schemes given in [16] and [67] is that the former was used for English handwriting and about 80% accuracy was got for the top-1 results and about 92% accuracy was got for top-10 results while the latter supported Arabic handwriting and its accuracy was 88% in top-1 and 99% in top-10 results.

In 2007, Vladimir Pervouchine et al. [38] implemented a writer identification scheme based on high frequent characters. In this method, the high frequent characters ('f','d','y','th') are first identified, and then according to the similarity of those characters, the writer is selected. The similarity is calculated with respect to the features such as height, width, slant, etc. associated with the characters. The number of features associated with each character is different (e.g. 'f' has 7 features while 'th' has 10 ones). A simple Manhattan distance was used in the classification phase. In order to select the best subset of the features, a GA(Genetic Algorithm) was used, which evaluated about 231 possible subsets, out of 5000 subsets. The system was tested in a database with 165 writers (between 15 to 30 patterns per writer), and the system accuracy was more than 95%. However, this method is simple and has good results, but the main concern of this method is that if a writer knows the procedure of

the method, he/she can write a text in test phase such that its characters are totally different with trained ones so that the system cannot identify him/her.

A major contribution by Bangy Li et al. [68], again in 2007, used the feature vector of hierarchical structure in shape primitives along with the dynamic and static feature for writer identification for 242 writers using NLPR online database and attained a result of above 90% for Chinese and about 93% for English. The substantiation given is that English text contains more oriental information than Chinese text. In 2008, Zhenyu He et al.[69], suggested an offline Chinese writer identification scheme which used Gabor filter to extract features from the text. They also incorporated a Hidden Markov Tree (HMT) in wavelet domain. The system was tested against a database containing 1000 documents written by 500 writers. Each sample contained 64 Chinese characters. The top-1, top-15, and top-30 results had 40%, 82.4%, and 100% accuracy, respectively [69]. Also, these authors have used a combination of general Gaussian model (GGD) and wavelet transform on Chinese handwriting[15]. They tested the method on a database gathered from 500 people. This database consisted of 2 handwriting images per person. In the experiments, top-1, top-15 and top-30 results had 39.2%, 84.8% and 100% accuracy, respectively. As, the authors reported, the accuracy of proposed methods was low especially in top-1.

In 2009, YuChenYan et al. [70] presented spectral feature extraction method based on Fast Fourier Transformation which was tested on the 200 Chinese handwriting text collected from 100 writers. The methodology showed 98% accuracy for top 10 and 64% for top1 using the Euclidean and WED classifiers. This scheme has the advantage of stable feature and also it reduces the randomness in Chinese character. Another advantage is that it is feasible for large volume of dataset. However it has higher computation

costs.

2.3.1 Arabic

Bulacu et al. proposed text-independent Arabic writer identification by combining some textural and allographic features [16], [71]. After extracting textural features (mostly relations between different angles in each written pixel) a probability distribution function was generated and the nearest neighborhood classifier was used for classification. For the allographic features, a codebook of 400 allographs was generated from the handwritings of 61 writers and the similarity of these allographs was used as another feature. The database in experiments consisted of 350 writers with 5 samples per writer (each sample consisted of 2 lines (about 9 words)). The best accuracy seen in experiments was 88% in top-1 and 99% in top-10. Also, a simpler definition of this method was presented by M. Bulacu et al. earlier in [22].

Also, Ayman Al-Dmour et al. designed an Arabic writer identification system in 2007 [72]. Different feature extraction methods such as hybrid spectral-statistical measures (SSMs), multiple-channel (Gabor) filters, and the grey-level co-occurrence matrix (GLCM) were verified to find the best subset of features. For the same purpose a support vector machine (SVM) was used to rank the features and then a GA (whose fitness function was a linear discriminant classifier (LDC)) chose the best one. Several classification methods such as LDC, SVM, weighted Euclidean distance (WED), and the K nearest neighbors (KNN) were also considered. The KNN-5, WED, SVM, and LDC results after feature selection per sub-images were reported as 57.0%, 47.0%, 69.0% and 90.0%, respectively. The results were better when the whole image was used, for instance the LDC result was increased to 100% (with no rotation). The database tested was gathered from 20 writers; each writer was asked to copy 2

A4 documents, one for training and the other one for testing. The used documents for each writer were different from the others and the sub-images were generated by dividing each document into 3x3 = 9 non-overlapping images. However, this method has good accuracy when LDC was used, but it seems the test database and samples per writer was small and it needs to be tested on more popular dataset. Faddaoui and Hamrouni opted for a set of 16 Gabor filters [73] for handwriting texture analysis. Gazzah and Ben Amara applied spatial-temporal textural analysis in the form of lifting scheme wavelet transforms. Angular features were considered as well in the task of Arabic writer identification [74].

Somaya Al-Ma'adeed et al. presented a text-dependent writer identification method in Arabic using only 16 words [34]. The features extracted include some edge-based directional features such as height, area, length, and three edge-direction distributions with different sizes and WED has been used as classifier. The test data was 32,000 Arabic text images from 100 people; the system was trained with 75% of the data and tested it by using 25%. They did not mention the top-1 accuracy of the method, but the best result in top-10 was 90% when 3 words were used. The main concern of this method is its dependency to text and the small dataset that were used in experiments. This method employed edge-based directional probability distributions, combined with moment invariants and structural word features, such as area, length, height, length from baseline to upper edge and length from base line to lower edge. On the other hand, Abdi et al. used stroke measurements of Arabic words, such as length, ratio and curvature, in the form of PDFs and cross-correlation transform of features [75] for the writer identification scheme.

Although, Arabic language is similar to Persian in character set and some writing styles, the Arabic methods cannot be extended to Persian language completely because of some special symbols that exists in Arabic

language.

2.3.2 Persian

In 2006, Shahabi et al. proposed a Gabor based system for Persian writer identification and the accuracy of their work was reported about 92% in top-3 and 88% in top-1[76]. It was observed that the testing was not adequate; because in the test phase, there was only one page per person such that 34 of it were used in training and the rest of page used in test phase. On retesting it, the method accuracy was of 60% in 80 people. In another scheme, Soleymani Baghshah et al. designed a fuzzy approach for Persian writer identification [42]. In this approach fuzzy directional features were used and the fuzzy learning vector quantization (FLVQ) recognized the writers. The drawback of this method is that it only works on disjoint Persian characters that are not conventional in Persian language. This system was tested using 128 writers and results were around 90%-95% in different situations of test.

A Persian handwritten identification system based on a new generation of Gabor filter called XGabor filter is proposed in [77]. Feature extraction was done using Gabor and XGabor filters. In the classification phase, weighted Euclidian distance (WED) classifier was used. The proposed method in [77] got 77% accuracy using PD100. Rafiee and Motavalli [78] introduced a new Persian writer identification method, using baseline and width structural features, and a feed forward neural network was used for the classification.
Another recent work has proposed an LCS (longest common subsequence) based classifier to classify features that are extracted by Gabor and XGabor filters [17], [79]. This classifier improved the system accuracy up to 95% on PD100. Even though, the features extracted by XGabor filter could model the characteristic of written documents, the accuracy of these

methods was not proper because of the problems in data classification and representation. Therefore, in the present paper, XGabor filter was used together with Gabor filter with different data representation, classification, and identification schemes. In another research, a mixture of some different methods has been used by Sadeghi ram et al. Grapheme based features are clustered by fuzzy clustering method and after selecting some clusters, final decision is made based on gradient features. The scheme got about 90% accuracy in average on 50 people that were selected randomly from PD100 [30].They also used a three layer MLP(multi layer perceptron) to classify the gradient based features, and they got about 94% average accuracy on same data set [80]. To the best of our knowledge, there is no other reported method in Persian writer identification. Table 2.1 summarizes the Writer Identification Methods on Multiple Languages. A graphical plot in Fig.2.3 compares the performance evaluation of different writer identification schemes across multiple languages.

Table 2.1: Writer Identification Methods on Multiple Languages

System	**Sample Space**	**Features**	**Classification Methodology**	**Accuracy**	**Language**
Text-dependent					
Srihari et al.s [1], [81]	1000 writers (CEDAR letter / paragraph / word)	Two levels of features; one at the macro level, micro level.	multi-layer perceptron	98%	English
Zois et al[33]	50 writers (45 samples of the same word)	The horizontal projection profiles are resampled into 10 segments, and processed using morphological operators	Bayesian classifiers and neural networks	95% for both English and Greek	English and Greek
Tomai et al. [32]	1000 writers	Character level, Word level features	Euclidean distances	99%	English
Zuo et al. [82]	40 writers	Offline PCA based method	Squared Euclidean distance	97.5%	Chinese
Zhang et al. [35]	*1000 writers*	Gradient (192 bits),structural (192 bits), and concavity (128 bits) features	k-nearest neighbor classification	97.71%	English
Somaya Al-Ma'adeed et al.[34]	*100 writers(320 words(16 different types))*	Height area, length and Edge –direction distribution	WED classifier	Top-10: 90%	Arabic

Writer Identification Methods on Multiple Languages

System	Sample Space	Features	Classification Methodology	Accuracy	Language
Text-dependent					
Schlapbach et al.[11]	200 writers(8 paragraph of about 8 lines)	Point-based (speed, acceleration, vicinity linearity, vicinity slope), stroke-based (duration, time to next stroke, number of points, number of up strokes, etc.),	Gaussian mixture model (GMM)	98.5%	English
Text-independent					
Pitak et al. [83]	81 writers	velocities of the barycenter of the pen movements	Fourier transformation approach	98.5%	Thai
Schlapbach et al. [84].	100 writers	X-Y coordinates	Hidden Markov Models	96%	English
Said et al. 19], T. Tan [54], Y. Zhu [55]	Two sets of 20 writers, 25 samples per writer (Few lines of handwritten text)	texture features using multichannel Gabor filtering and gray-scale co-occurrence matrices	Nearest centroid classification using weighted Euclidean distance	96%	English
Bensefia et al. [24], [59], [60], [61]	88 writers (French), 150 writers (English)	A textual based Information Retrieval model, local features such as graphemes extracted from the segmentation of cursive handwriting	Cosine similarity	95% on 88 writers 86% on 150 writers	French /English

Writer Identification Methods on Multiple Languages

System	**Sample Space**	**Features**	**Classification Methodology**	**Accuracy**	**Language**
S. K. Chan [85]	82 writers	x-y coordinates, direction, curvature of x-coordinates and the status of pen up or pen down.	Discrete Character prototype distribution approach (Euclidean distance)	*95%*	*French*
Marti et al. [28] and Hertel and Bunke [27]	20 writers (5 samples of the same text)	Height of the three main writing zones, the distances between connected components	k-nearest neighbor and a feed forward neural network classifier	90%	English
M. Bulacu [22],[23],[86],[87]	650 writers	Edge based directional PDFs as features (Textural and allograph prototype approach)	k-nearest neighbor and a feed forward neural network classifier	92%	English
Guo Xian Tan Christian[29]	120 writers	Continuous Character prototype distribution approach	Minimum distance classifier	99%	French
Neils et al.[88]	43 writers	Allograph prototype matching approach using the dynamic time warping (DTW) distance function	*af-iwf (allograph frequency – inverse writer frequency) measure*	60%	English

Writer Identification Methods on Multiple Languages

System	**Sample Space**	**Features**	**Classification Methodology**	**Accuracy**	**Language**
B. Helli, et al. [17], [71] [79]	100 writers (PD100 dataset), 50 writers[46]	Point-based (speed, acceleration, vicinity linearity, vicinity slope), stroke-based (duration, time to next stroke, number of points, number of up strokes, etc.).	Tey proposed an LCS (longest common subsequence) based classifier	95%	Persian
Bangy Li et al. [68]	242 writers*(*NLPR online handwriting Database and 50 Chinese and English words in one page*)*	Hierarchical Structure in Shape Primitives + Fusion Dynamic and Static Features	nearest neighbor classifier	Chinese accuracy>90% English accuracy>93%	English and Chinese text
YuChen Yan et al. [70]	200 handwritings from 100 writers	Spectral feature based on Fast Fourier Transformation	Euclidean and WED classifiers	98% -top 10 64%-top1	Chinese

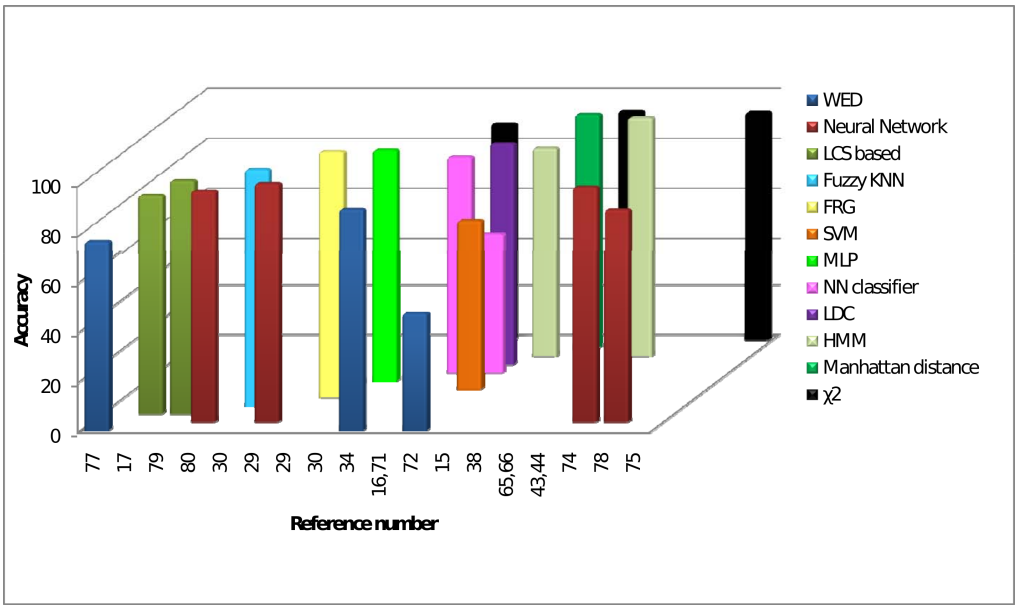


Figure 2.3: comparative evaluation of writer identification schemes

2.4 Writer Identification in Indian Languages

Very few studies in Indian Languages have been documented so far. Table 2.2 illustrates the research work done in the area. Currently the writer identification of handwritten Malayalam documents is done manually. In the preliminary analysis, with a global overview, physiological character shape formation is considered. In this, appearance of a character and its special characteristics like slant angle, slant directions etc are taken into account. Making it further, in detailed analysis minute features like writer specific allographs or discriminating characters, height-width ratio, distance/space between character/words, applied pressure in writing process etc are considered.

Handwriting analysis can be considered as a behavioral biometric system. This calls for multilevel observations. Hence our work progresses from grapheme level to character level and then to document level in a way

to achieve our goals. The research methodology is discussed in detail from chapter 3 onwards. It includes four major tasks as given in Fig 2.4. Four classifiers such as Naive bayes, k-NN, SVM and Adaboost M2 are used for identification in the first three phases. Inorder to find the decisive feature for Malayalam characters, different features are extracted from each phase.

- **Phase 1**: The grapheme level features such as directional features, Curvature and Angle pair features are used.
- **Phase 2**: Character level features like loop features, directional features, distance features and geometrical features are considered.
- **Phase 3**: WD-LBP and SIFT features are considered for the document level.
- **Phase 4**: The efficiency, consistency and stability of the features are analyzed in each of the three phases. This is further used for the two applications such as online character recognition of Malayalam and Grantha scripts

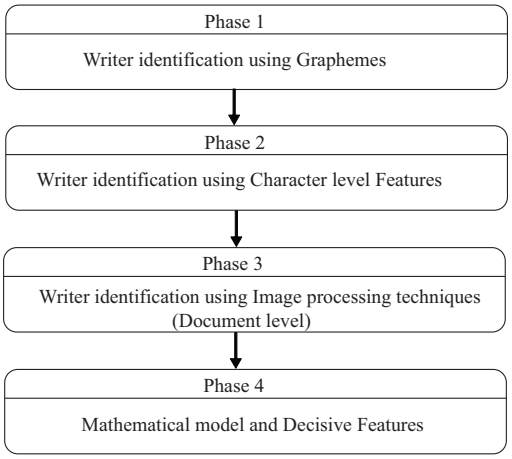


Figure 2.4: Development phases in this research work

Table 2.2: Writer Identification methods on Indian Languages

System	**Sample Space**	**Features**	**Classification Methodology**	**Accuracy**	**Language**
Text-dependent					
B.V.Dhandra and Mallikarjun Hangarge *et.al*[153]	250 writers.	Potential visual discriminating features are extracted as global and local features	KNN classifier	The proposed algorithm achieves an average maximum recognition accuracy is 96.05% and 99% respectively for text words and numerals with five fold cross validation test.	Kannada, Roman (English), Devanagari

Writer Identification methods on Indian Languages

System	**Sample Space**	**Features**	**Classification Methodology**	**Accuracy**	**Language**
Pulak Purkait, Rajesh Kumar and Bhabatosh Chanda *et.al*[154]	22 writers	directional opening, directional closing, directional erosion and k-curvature features.	nearest neighbor	Merely 5 words in combination gives an accuracy of more than 90% with each of the four feature sets.	Telugu
Utpal Garain and Thierry Paquet *et.al*[155]	RIMES containing 382 French writers and ISI consisting of 40 Bengali writers.	*2D AR model coefficients*	Euclidean distance	62.1% for combined dataset	French, Bengali

2.5 Summary of the chapter

Literature survey has enabled to see that a wide variety of features are used for writer identification. In Chinese language, writer-specific texture features using multichannel Gabor filtering and Gray-Scale Co-occurrence Matrices are common, but in English it varies from micro level features to macro level and edge distribution. Also studies are carried out in other languages like Arabic and in Persian. Combinations of some textural and allographic features, hybrid spectral-statistical measures (SSMs), multiple-channel (Gabor) filters, XGabor etc. are carried out to obtain the individuality of the writers. Also studies show that these features when applied to other languages achieved only lesser accuracy. From this we understand that features must be selected based on the characteristic features of each language.

From the discussion of text-dependent and text-independent methods, we can conclude that in general, higher identification rates are achievable with the former type of text-dependent methods. Where as Text-independent methods are much more useful and applicable. These methods, however, require a certain minimum amount of text to produce acceptable results. We could say that the research on writer identification that started with the analysis of very constrained writings and very few writers has matured really well over time. Regarding the methods developed, in addition to the structural and statistical features, codebook generation has emerged as a very popular as well as effective method for writer identification. These codebooks could be computed universally for the entire set of writers or for each of the writers separately. The methods based on a universal codebook are generally efficient in terms of computational cost, however, a new codebook is to be generated if the script changes. On the other hand, writer specific codebooks have

high computational costs but they could present a generic framework independent of the alphabet under study. In the writer identification methods discussed here the features are independent of the textual content of each language.

Chapter 3

Writer Identification using Graphemes

This chapter describes a novel approach to writer identification in Malayalam using graphemes. Graphemes are small writing fragments extracted from the handwritten documents which contain meaningful patterns and possess individuality of each writer. Different classifiers like Naive-Bayes, k-NN, SVM and Adaboost were experimented with a comparative evaluation of different classifiers is done. Also the identification rate for different features under consideration is computed, thus aiding to find out the influential feature. Different methods to

eliminate redundant characters and their influence are done in the architecture. Experiments were done to draw conclusions at the influence of amount of text present for each writer and the codebook size on the identification rate for the different classifiers.

3.1 Introduction

In the previous chapter, the significant contributions to the field of writer recognition over the last two decades were discussed. However by the beginning of the year 2000 A.D the focus of research in the domain started shifting towards the extraction of writer specific patterns in writing. Graphemes are fragments or small parts of handwritten text which are obtained by the segmentation of handwriting into small windows. In Malayalam script, it is observed that the majority of symbols are formed by loops and curves. An individual who draws a particular loop or curve in a specific way is expected to always employ the same pattern when drawing that shape, irrespective of the character being written. Graphemes do have the capacity to capture this, if the fragment size is chosen properly and they can be characterized as the individuality of the writer. Hence graphemes are chosen here for performing writer identification.

In Section 3.2 a scheme of the writer identification using graphemes which is suitable for Malayalam language is given. Section 3.3 briefs on the corpus available for testing. Section 3.4 portrays the implementation details. Section 3.5 analyses the results obtained and provide valid conclusions. The chapter is concluded in Section 3.6.

3.2 Scheme Design

The writer identification system generally consists of a training phase and an identification phase. The system design is given in Fig. 3.1 and the modules of it are described in the subsequent sections.

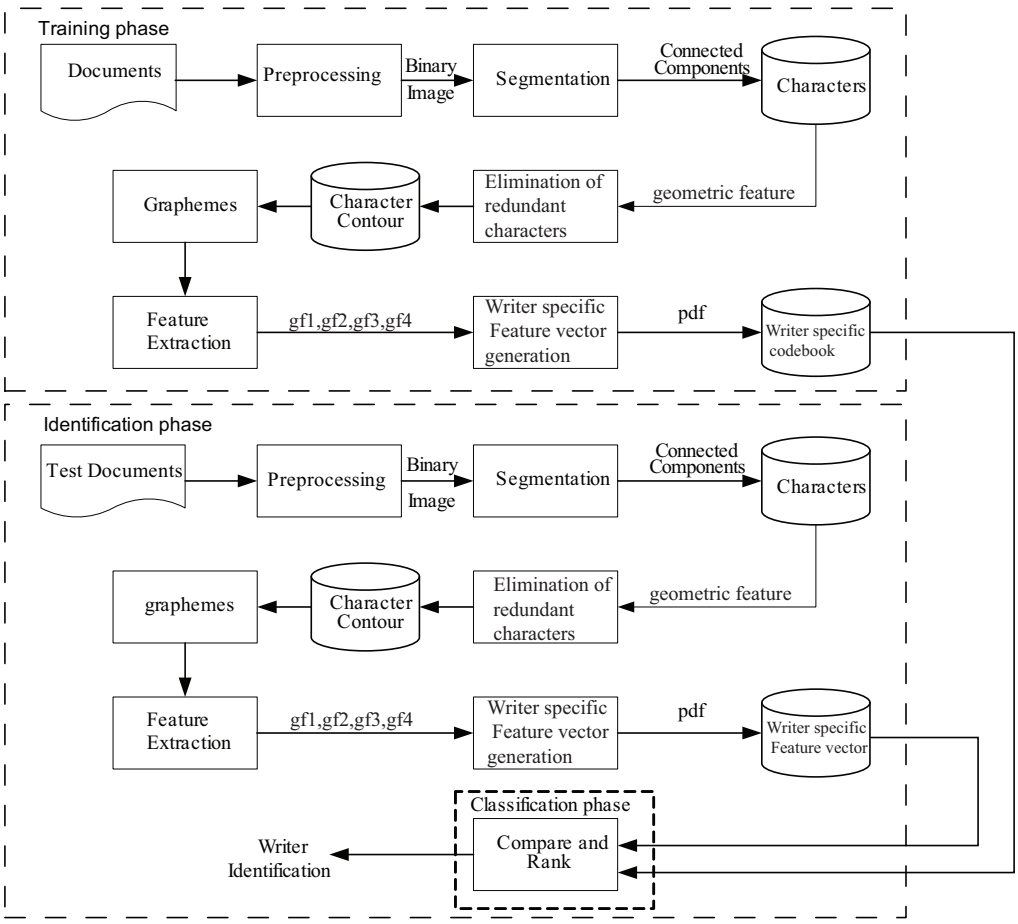


Figure 3.1: Schematic diagram of the system

3.2.1 Preprocessing

The images of the documents are processed in order to obtain the fragmented connected components representing each Malayalam character. For this, images are scanned at the same resolution and the spurious data/ noise is removed from the native data for further processing. Usually for writing instrument independency, the distribution of ink widths in the validation data set are examined and normalized for each writings to a fixed thickness using a thinning algorithm. However this resulted in a degradation of performance as some writer-specific information is also lost during the normalization. Hence in this work it was decided not to normalize the ink thickness. The naive image of documents should be pre-processed at first hand through the following steps.

- Convert the scanned document into 8-bit gray scale image.
- Calculate the mean intensity and standard deviation of the gray scale image.
- Find out a threshold value based on the mean and standard deviation such that points above the value can be classified as white and others as black.
- Convert the gray image into binary image around the above threshold value.
- Image de-noising is practiced to attain the perfect binary image of the documents [72][89].

3.2.2 Segmentation

The preprocessed document is segmented into words using RLSA [102] and Recursive XY Cuts [103] methods. Words are split into characters using

the standard connected components algorithm [104]. Malayalam has no cursive form of writing. So each connected components can be considered as a character. For each connected component, its contour was computed using Moore's algorithm starting at the left most pixels in clockwise fashion because of the writing form of Malayalam and the resulting contour of each character was computed. Fig 3.2 shows the example of the connected component of the Malayalam character 'ക' (ka)

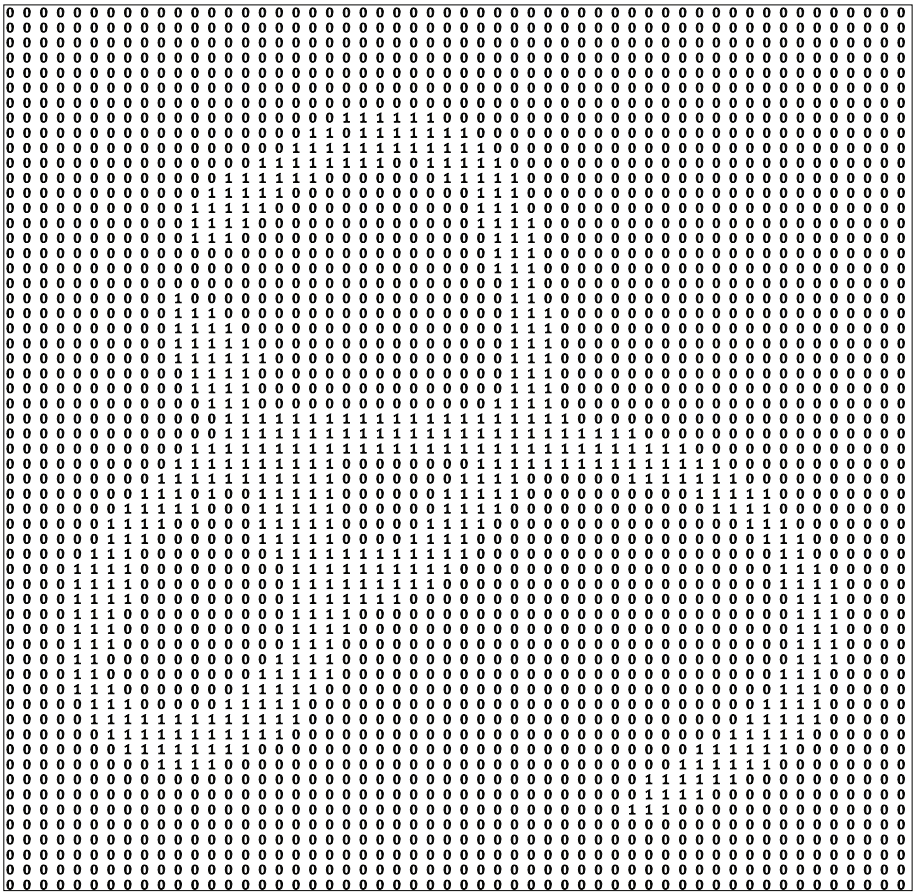


Figure 3.2: connected component of the character 'ക' (ka)

3.2.3 Elimination of redundant characters

The unique characters of each writer are to be found out for individuality identification and they aid in classifying the writers. Redundant characters which have same style are to be eliminated. The following algorithm is used for the elimination of redundant characters

Algorithm 1 Elimination of redundant characters

1. Let there are 'n' characters in the whole set of extracted characters/connected components.
2. For each character C_i, compute elliptic feature vector (section 4.3.4) f_i, where $i = 1\ to\ n-1$
3. Take each character C_i
4. For all characters C_j where $j = i+1\ to\ n$
5. Compute the dissimilarity between C_i and C_j as,
6. $Dissimilarity(C_i, C_j) = Distance(fi, fj)$
7. Go to step 3 until all $C_i s$ are inspected.
8. If the dissimilarity between C_i and C_j is less than threshold 0.05, then C_j is redundant with respect to C_i and hence eliminate C_j.

The complexity of this algorithm is $O(n^2)$. Van der maaten et.al [63] had shown that random selection can reduce the size of the feature space. However the method of elimination of redundant characters given in Algorithm 1 provided better accuracy in writer identification and the comparison is given in section 3.5.

3.2.4 Graphemes

The splitting of handwriting is an important step, since a 'good' splitt would allow exploiting the redundancy in writing. For our problem, a

'good' split is the one that yields writing fragments in a meaningful manner during comparison of these fragments. The split is done by using square windows of size n. The value of n should be large enough to contain plentiful information about the style of the writer and small enough to guarantee a good identification performance. For our system, the window size was fixed to 11x11. The splitting is carried out according to the sliding window positioning algorithm as mentioned in [90].After applying the segmentation each character is transformed into several fragments called graphemes. Fig 3.3 depicts the stages of creating graphemes of character 'ക' (ka). Fig 3.3 (a) is the character 'ക' (ka), Fig 3.3 (b) shows the sliding window position over the character 'ക' (ka) and Fig 3.3 (c) shows the corresponding graphemes.

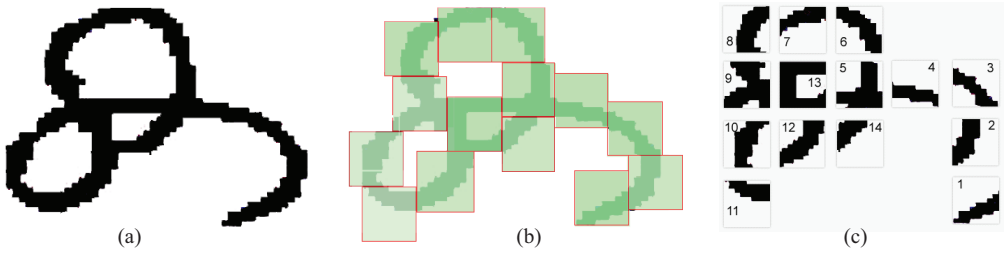


Figure 3.3: Generation of graphemes of the character 'ക' (ka)

3.2.5 Characteristic Features of Graphemes

Each grapheme is characterized by the following features.

Directional features (*gf1* & *gf2*)

Directional features of a grapheme are to be analysed at two levels. One is the chain code pair direction of the contour and the other is the

local direction along the contour (local stroke direction) to obtain the individuality of a writer.

Directional feature using chain code pairs(*gf1*)

In the case of handwritten document images, directional features and their improved variants have been successfully applied to character/word recognition [90] [91] [92] [93] [94] as well as classification of writing styles [95]. Although a peripheral idea can be obtained from the slope histogram on writer specific character shapes, it is insufficient to explain detailed variants in the writing style. At the global level, the overall orientation information in the writing can be confined to the contribution of the eight principal directions in an individual's writing. Fig 3.4(a) shows the eight principal directions. In order to confine the finer details; we have to count not only the occurrences of the individual chain code directions but also the chain code pairs. The $bin(i, j)$ of the (8x8) histogram given in Fig 3.7 represents the probability of finding the pair(i,j) in the chain code sequence of the contours. For extracting the feature *gf1* the four types of L-junction can be considered as shown in Fig 3.4(b). To obtain the bin values, L-junctions were found at the contour traversal. For each L-junction, we can have only one of the two possible chain code pair due to the difference in orientation of the contour (clockwise/ anti-clockwise). Fig 3.5 shows the possible chain code pairs of the first L-junction shown in Fig 3.4(b). These pairs are dependent in the direction of contour tracing and corresponding chain code sequence.

To explain further Fig 3.6 (a) shows the eight principal directions, Fig 3.6 (b) shows the character ৩ (ra), Fig 3.6(c) shows its contour, Fig 3.6(d) represents one grapheme of Fig 3.6(c).

The normalized histogram of the directional distribution feature *gf1* is given in Fig 3.7.

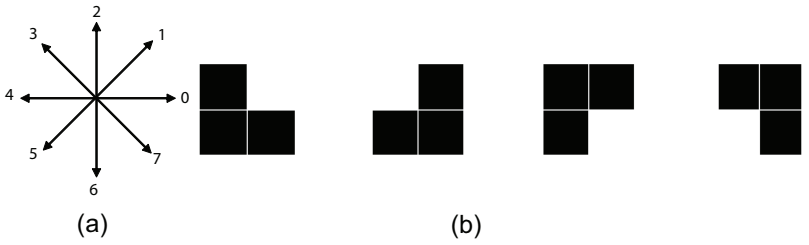


Figure 3.4: Four possible L-junctions

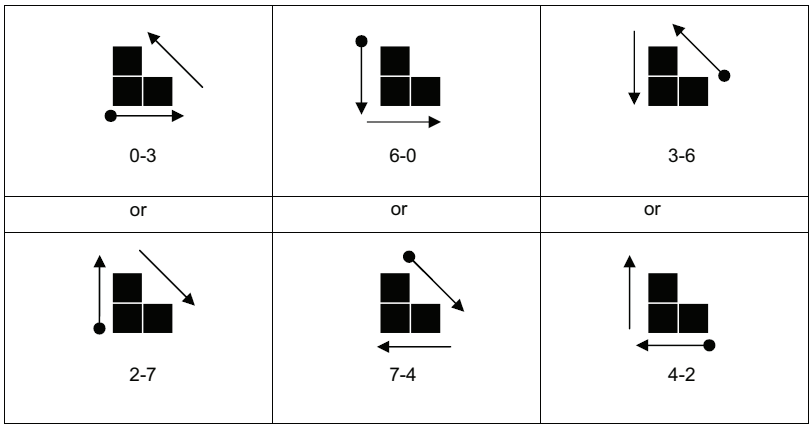


Figure 3.5: Possible chain code pairs, starting at each of the three pixels of L-junctions

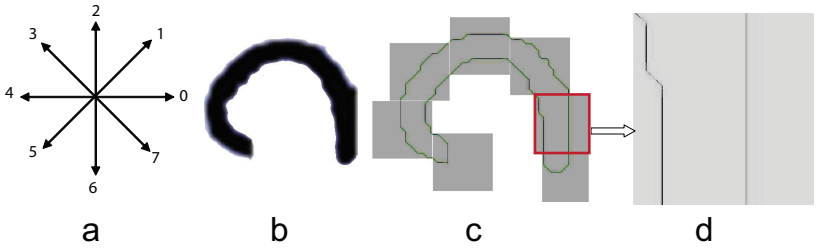


Figure 3.6: Processing stages of a grapheme

The chain code sequence corresponding to grapheme in Fig 3.6(d) is

Table 3.1: Chain Code Sequence

Chain code sequence:-	7	6	6	6	7	7	6	6
Chain code pair:-	7-2	6-1	6-1	6-1	7-2	7-2	6-1	6-1
Chain code sequence:-	6	6	6	6	0	0	0	0
Chain code pair:-	6-1	6-1	6-1	6-1	0-3	0-3	0-3	0-3
Chain code sequence:-	2	2	2	2	2	2	2	2
Chain code pair:-	2-5	2-5	2-5	2-5	2-5	2-5	2-5	2-5
Chain code sequence:-	2	2	2	2	4	4	4	4
Chain code pair:-	2-5	2-5	2-5	2-5	4-7	4-7	4-7	4-7
Chain code sequence:-	4	4						
Chain code pair:-	4-7	4-7						

Table 3.2: Direction Matrix is

Directions	0	1	2	3	4	5	6	7
0				4				
1								
2						12		
3								
4								6
5								
6		9						
7			3					

Local stroke direction(*gf2*)

This mainly aims at obtaining the writing flow of a writer. The whole text is divided into small graphemes in uniform manner in a view to extract the contour of the text. The analysis is similar to one described in the previous subsection of chain code pairs. The first step involved in the extraction of the feature *gf2* is calculating the chain code sequence of each

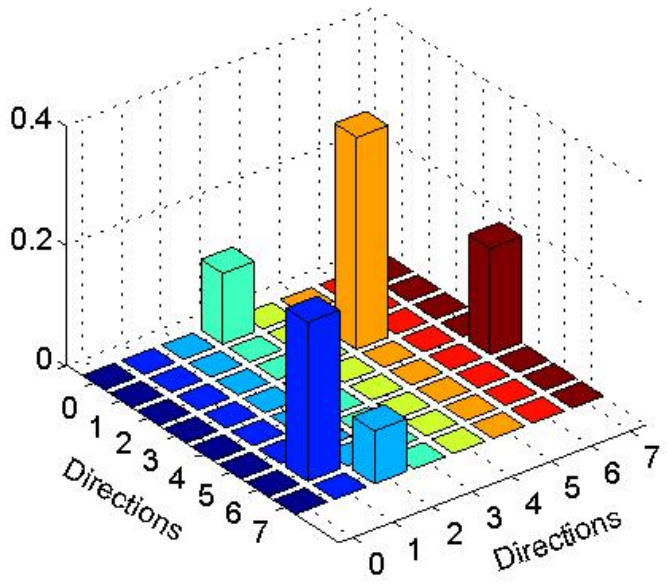


Figure 3.7: Normalized histogram of direction distribution feature *gf1* corresponding to the grapheme in Fig 3.6(d).

grapheme. Let the chain code sequence of a contour in a window is denoted by CS^w. The contour length within the window has been determined with the distribution of chain code connectivity. The eight directions d (Fig 3.6(a)) and the number of intervals p are accumulated by a dXp array. The percentage of the total contour length divided into 'p' intervals should be large enough to capture the differences in the distributions of eight directions within the window. 'p' is set to be 10 from the variations of 2 to 20 on the validation data set. The accumulator is initialized with all bins set to zero. The $bin(i, j)$ of the accumulator is incremented by 1 for each window w, containing the chain code sequence CS^w, if the frequency

of direction i is represented in the j^{th} interval, where j is given by

$$j = ceil\left(\frac{cardinality(CS_i^w)}{cardinality(CS^w) \times p} \times 100\right) \; with \tag{3.1}$$

$$(CS_i^w) = \{CS_k \in CS^w | CS_k = i\} \; and \; i = 0, 1, ...7 \tag{3.2}$$

Consider the example of character ಠ (ra). To determine the local direction first we determine the number of pixels corresponding to each direction from the chain code sequence. Next the percentage of these pixels in the contour length is calculated. Further the local distribution of these pixels is obtained by finding the interval of each direction using equation 3.1. The process has been illustrated in Fig 3.8 and the normalized stroke direction histogram of the selected grapheme in Fig 3.6(c) is illustrated in Fig 3.9 where the five directions encountered in the window results in incrementing the respective bins of the distribution *gf2*.

The Chain code sequence corresponding to grapheme in Fig 4.6(d) is
7 666 77 666 666 0000 222 222 222 222 444 444

Number of pixels in each direction

Directions	0	1	2	3	4	5	6	7
Pixels	4	0	12	0	6	0	9	3
Percentage of Contour length	11.765%	0%	35.294%	0%	17.640%	0%	26.470%	8.824%
Interval	2	0	4	0	2	0	3	1

Figure 3.8: Analytical process of the distribution of *gf2*

Curvature (*gf3*)

The curvature of a grapheme is defined as $K(s) = \lim_{h \to 0} \frac{\phi}{h}$,where $\phi$is the angle between $\mathbf{t}(s)$ and $\mathbf{t}(s + h)$. $\mathbf{t}$ represents the tangent vector and s is the arc length parameter. *Curvature-zero crossings* of a curve

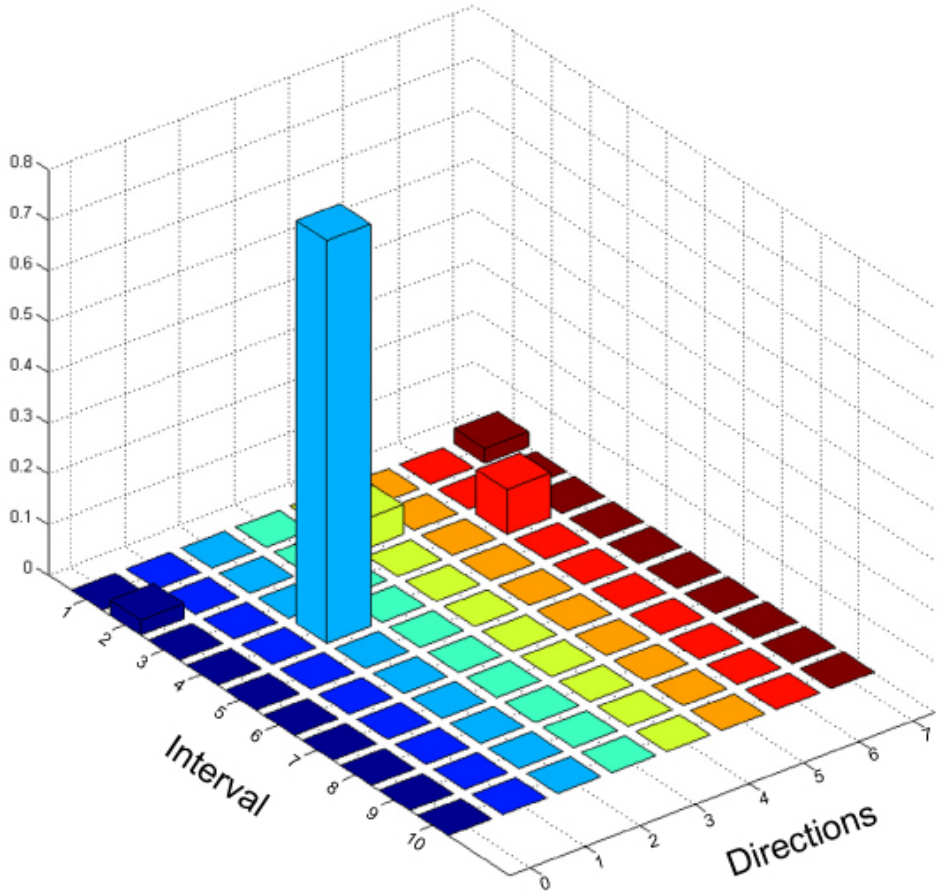


Figure 3.9: Histogram of the normalized stroke direction distribution feature *gf2* for the grapheme in Fig 3.6(d).

are points where the sign of curvature changes. Consider a parametric vector equation for a curve: $\Gamma(u) = (x(u), y(u))$where u is an arbitrary parameter. The formula for computing the curvature function can be expressed as

$\kappa(u) = \frac{\dot{x}(u)\ddot{y}(u) - \ddot{x}(u)\dot{y}(u)}{(\dot{x}^2(u) + \dot{y}^2(u))^{3/2}}$

If we convolve each component of Γ with $g(u, \sigma)$a 1D Gaussian kernel of widthσ, then $X(u, \sigma)$and $Y(u, \sigma)$represent the components of the resulting curve, Γ_σ :

$$X(u, \sigma) = x(u) * g(u, \sigma) \tag{3.3}$$

$$Y(u, \sigma) = y(u) * g(u, \sigma) \tag{3.4}$$

According to the properties of convolution, the derivatives of every component can be calculated easily:

$$X_u(u, \sigma) = x(u) * g_u(u, \sigma) \tag{3.5}$$

$$X_{uu}(u, \sigma) = x(u) * g_{uu}(u, \sigma) \tag{3.6}$$

and we will have a similar formula for $Y_u(u, \sigma)$and $Y_{uu}(u, \sigma)$ Since the exact forms of $g_u(u, \sigma)$and$g_{uu}(u, \sigma)$ are known, the curvature on $\Gamma_\sigma$can be computed easily:

$\kappa(u, \sigma) = \frac{X_u(u,\sigma)Y_{uu}(u,\sigma) - X_{uu}(u,\sigma)Y_u(u,\sigma)}{\left(X_u(u,\sigma)^2 + Y_u(u,\sigma)^2\right)^{3/2}}$

The process starts with $\sigma = 1$, and at each level, σ is increased by$\Delta\sigma$, chosen as 0.1 in our experiments. As σincreases, $\Gamma_\sigma$shrinks and becomes smoother, and the number of curvature zero crossing points on it decreases. So we kept the value of $\sigma = 1$ because of not destroying the individuality of the writer.

The curvature of each grapheme is calculated using the above method. The value of *gf3* corresponding to grapheme in Fig 3.6(d) is computed as the average of the PDF(Probability Density Function) values given in Table 3.3.

Feature vector corresponding to the above PDF of curvature is 0.119833

Table 3.3: Curvature (*gf3*)feature and its PDF

curvature	0.02	0	0	0.01	0.03	0.14
pdf	0.131467	0.131189	0.131189	0.131329	0.131604	0.133015
curvature	0	0	0	0	0	10.07
pdf	0.131189	0.131189	0.131189	0.131189	0.131189	0.000921
curvature	0.07	0.01	0.01	10.03	0	0
pdf	0.132137	0.131329	0.131329	0.000962	0.131189	0.131189
curvature	0	0	0	0	0	0
pdf	0.131189	0.131189	0.131189	0.131189	0.131189	0.131189
curvature	0	0	0	10.13	0.01	0.01
pdf	0.131189	0.131189	0.131189	0.000863	0.131329	0.131329
curvature	0	0	0.01	0.03		
pdf	0.131189	0.131189	0.131329	0.131604		

Angle pair (*gf4*)

The Edge-hinge distribution is a feature that characterizes the changes in the direction of a writing stroke in handwritten text [23]. The edge-hinge distribution can be extracted by means of a window that is sliding over an edge-detected binary handwriting image. The goal of this method is to generate a feature characterizing the changes in direction undertaken during writing with the hope that it will be more specific to the writer and consequently making possible more accurate identification. The central idea is to consider in the neighbourhood two edge fragments emerging from the central pixel and, subsequently, compute the joint probability distribution of the orientations of the two fragments. To illustrate this, consider a hinge laid on the surface of the image. Let its junction be placed on top of every edge pixel, then open the hinge and align its legs along the edges. Consider then the angles ϕ_1 and ϕ_2 that the legs make with the horizontal and count the found instances in a two dimensional array of bins indexed by ϕ_1 and ϕ_2 as shown in Fig 3.10. The final normalized

histogram gives the joint probability distribution p(ϕ_1 and ϕ_2); quantifying the chance of finding in the image two "hinged" edge fragments oriented at the angles ϕ_1 and ϕ_2 .

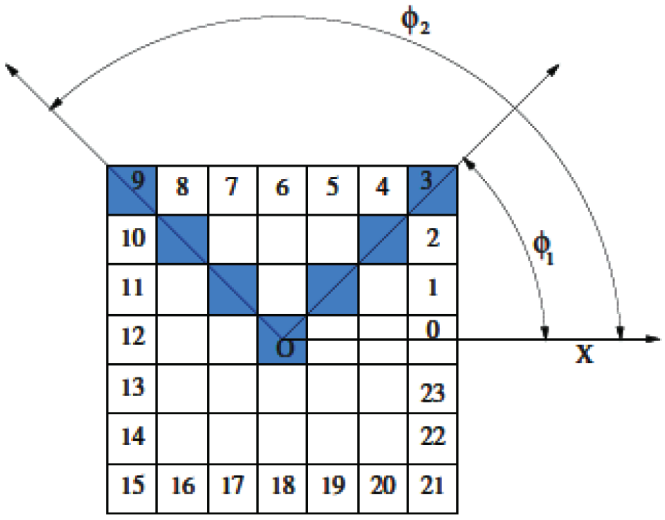


Figure 3.10: Edge Hinge Distribution[23]

Edges are usually wider than 1-pixel and therefore an extra constraint is imposed that the ends of the hinge legs should be separated by at least one "non-edge" pixel. This makes certain that the hinge is not positioned completely inside the same piece of the edge strip. This is to make sure that feature properly describes the shapes of edges and avoids the senseless cases. The orientation is quantized in 2n = 16, 24 and 32 directions respectively (Fig. 3.10 is an example for 2n = 24). From the total number of combination of two angle we will consider only the non-redundant ones ($\phi_2 > \phi_1$) and we will also eliminate the cases when the ending pixels have a common side. However in this system a complete 2D probability distribution that takes into account all possible combinations of angles pairs as the feature *gf4* is considered. The value of *gf4* corresponding

to grapheme in Fig 3.6(d) is computed as the joint PDF values given in Table 3.4. Feature vector corresponding to the angle pair is computed as the average of the PDF values given in Table 3.4 which corresponds to 0.597315.

3.2.6 Codebook Generation

Codebooks are generally categorized as writer specific and universal. In the first type, for each writer, the redundant patterns are extracted separately, whereas in the second type the redundant patterns are extracted from the global data set. There are numerous writer identification schemes available based on these types of codebooks [22][23][24][25]. While generating a codebook of the characteristic writings, we group them into classes represented by a set of features. In grapheme level writing fragments of an individual may be grouped into clusters in a variety of ways. Methods like k-means, fuzzy c-means, learning vector quantization and the closely related self organizing maps have been successfully applied to similar problems of clustering allographs or graphemes [29] [96]. Each writer generates a finite number of basic patterns (graphemes) and each pattern is characterized by four parameters namely directional features (*gf1* & *gf2*), curvature (*gf3*) and angle pair (*gf4*). These basic patterns are clustered using Kohonen self-organizing feature map (SOM 2D). While varying the network size from 2 x 2 to 50 x 50 it was found that 11x11 network gave optimal performance. The corresponding graph is shown in Fig 3.14. Next the PDF of each cluster and its average is calculated to obtain the writer specific feature vector and it is stored as a writer specific codebook entry. Hence the size of the writer specific feature vector was fixed as 121 (11x11). This writer specific feature vector was used for the generation of the codebook. In our study we have considered 280 writers; hence the codebook size is 280 and each entry with 121 feature dimension. A fragment

Table 3.4: Angle pair (*gf4*)feature and its PDF

Cos ϕ_1	Cos ϕ_2	pdf (ϕ_1)	pdf (ϕ_2)	joint pdf(ϕ_1,ϕ_2)
1	0.7071	0.026734	0.420824	0.01125
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0.7071	0.831532	0.420824	0.349929
-0.7071	0.5736	0.439071	0.645467	0.283406
-0.5736	0	0.601803	0.931137	0.560361
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	1	0.831532	0.104648	0.087018
-1	1	0.160037	0.104648	0.016748
-1	1	0.160037	0.104648	0.016748
-1	1	0.160037	0.104648	0.016748
-1	0.1737	0.160037	1.072183	0.171589
-0.1737	0	0.899618	0.931137	0.837668
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
-1	0	0.160037	0.931137	0.149017
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427
0	0	0.831532	0.931137	0.77427

of the codebook is given in Appendix A.

3.3 Dataset-Malayalam Handwritten Document Corpus(MHDC)

Absence of dataset of handwritten pages of different writers in Malayalam posed a great challenge in implementing the scheme. Hence, we collected the handwritings of 280 different users of similar as well as variable content of Malayalam text. The images have been scanned at 400 dpi, 8 bits/pixels, gray-scale. A total of 280 writers contributed to the data set with 50 writers having only one page, 215 writers with at least 2 and 15 writers with at least 4 pages. Each page consists of 21 lines of words with a minimum 30 characters in each line. We kept only the first two images for the writers having more than two pages and divided the image into two parts for writers who contributed a single page thus ensuring two images per writer, one used in training while the other in testing.

3.4 Implementation

As shown in the scheme design (Fig 3.1) the system consists of mainly two phases, training phase and an identification phase.

Training Phase

In this phase training dataset would be preprocessed without losing its individuality. The binary image obtained through preprocessing was put for segmentation in order to obtain connected components, in case of Malayalam, the characters. This connected component analysis is straight-forward, efficient and easy to implement for the segmentation of each character. This would figure up a character database. Elliptic features

(a,b,θ and count) (section 4.3.4) were used to eliminate the redundant characters. The contour of the set of distinctive characters obtained by using Moor's algorithm was put into graphemes. Directional features (*gf1* & *gf2*), curvature (*gf3*) and angle pair (*gf4*) features were extracted from the graphemes and clustered using kohonen SOM to generate a writer specific codebook.

Identification Phase

In this phase, a query descriptor (test document) pass through the modules in training phase where the features are extracted and a feature vector corresponding to the query descriptor is obtained. This would be classified with writer specific codebook of trained documents. Prepare a sorted hit list with increasing distance value (similarity score) between the query descriptor and writer specific codebook. Select the first ranked sample which will ideally identify the writer.

3.5 Experimental results

The primals of each writer have to be obtained for the sake of individuality identification. It is needed to avoid the redundant characters which have same style in his/her documents. This can also result in the reduction of the feature space. Van der maaten et.al [63] had improved the writer identification by random selection. This method was adopted for Malayalam characters for eliminating redundant character phase. For the elimination of redundant characters three feature methods were tried. One based on the width/height ratio & the curvature of the character, the second one based on the elliptic feature parameters (θ, *a and b*) (section 4.3.4) and the third one based on the random selection [63]. It was observed that the implementation of elliptic features gave best accuracy among the

width/height ratio & the curvature and randomized selection method. This is due to low allographic variation in Malayalam. Fig 3.11 depicts the comparison of the three methods. For the similarity measurement between characters three distance measures were tried. They were Euclidean, chi-square and Manhattan. It was observed that Chi-square was much superior to other methods as the number of writers increased. Fig 3.12 shows the variations in the identification rate as the number of writers increases with respect to the three distance measures.

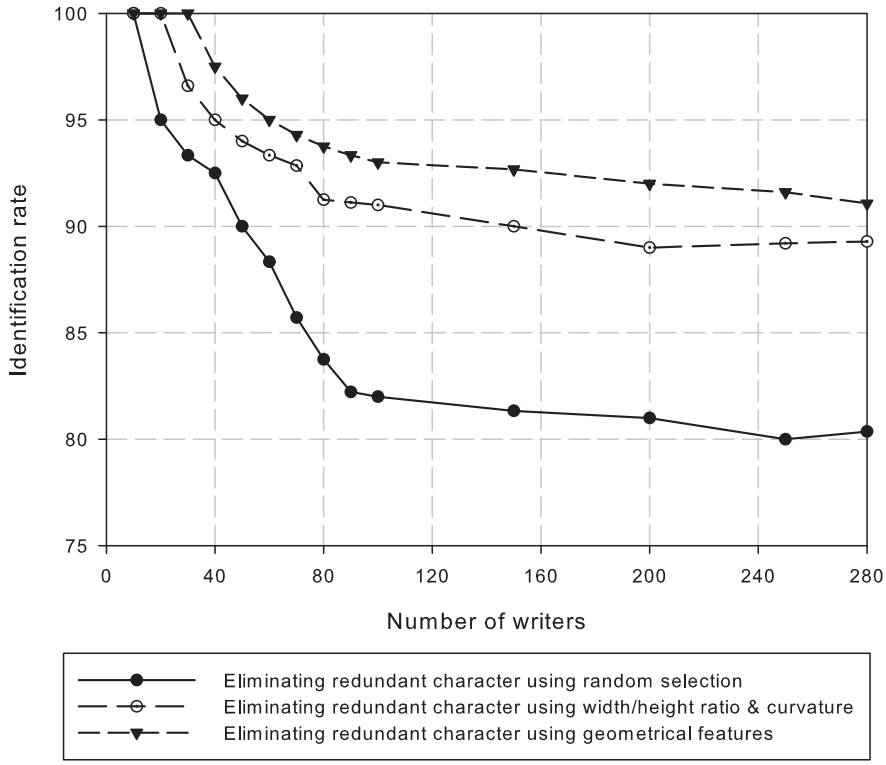


Figure 3.11: Comparative result of methods used for elimination of redundant characters

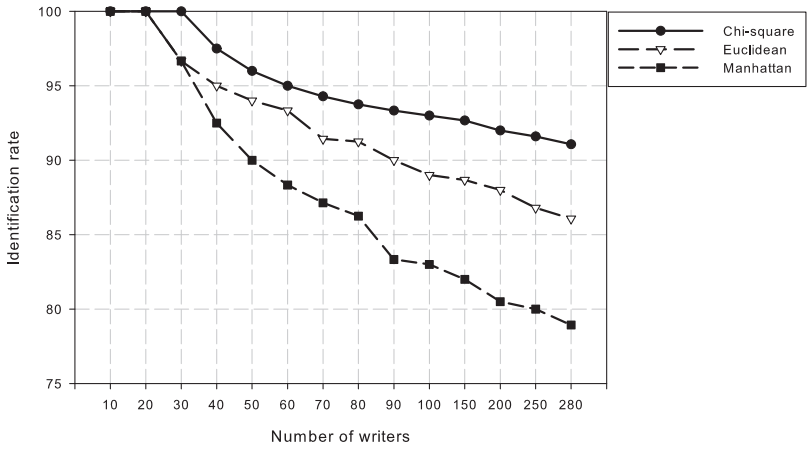


Figure 3.12: Identification rates of various distance measures used in the elimnation of redundant characters

The influence of each of the four parameters of the feature vector on the accuracy of identification was also carried out. Table.3.5 represents the potentiality of each of the features. It can be seen that accuracy of classifier can be very much increased by combining the feature rather than keeping them alone. Also the performances of different types of classifiers like Naive bayes, k-NN, SVM and Adaboost M2 were carried out. Table 3.6 gives the accuracy of identification rate for different classifiers. To detect the influence of amount of text used for each writer as dataset for the identification rate, further experiments were tried. The text was divided zone wise into eight zones(one word - two words; one line - two lines - three lines - four lines; one paragraph; full page) varying from one word to one full page. The result is summarized in Fig 3.13. It is evident that the Adaboost M2 is relatively more stable when the amount of text is varied where the identification rate rises from about 15.71% for a single word to 91.71% for the complete page. From the graph it is also evident that the

Naive-Bayes is very less sensitive to the amount of text used.

Table 3.5: Comparative evaluation of features

Number of writers	*gf1*	*gf2*	*gf3*	*gf4*	Combined feature *(gf1+gf2+gf3+gf4)*
10	70	80	90	100	100
25	68	72	88	96	100
50	62	72	86	96	98
75	61.33	69.33	85.33	96	96
100	60	68	85	90	93
150	60	66	84	88	92
200	59	65.5	83	86	91.5
250	58.8	64	82.4	85.6	91.6
280	58.2	63.2	82.14	84.29	91.79

Table 3.6: Recogntion Rate for different classifiers

Number of writers	Naive bayes	k-NN	SVM	Adaboost M2
10	100	100	100	100
25	92	96	96	100
50	90	94	96	98
75	89.33	93.33	94.67	96
100	88	91	92	93
150	86.67	90	91.33	92
200	85.5	89	90.5	91.5
250	85.6	89.2	90.4	91.6
280	85.71	89.29	90.71	91.79

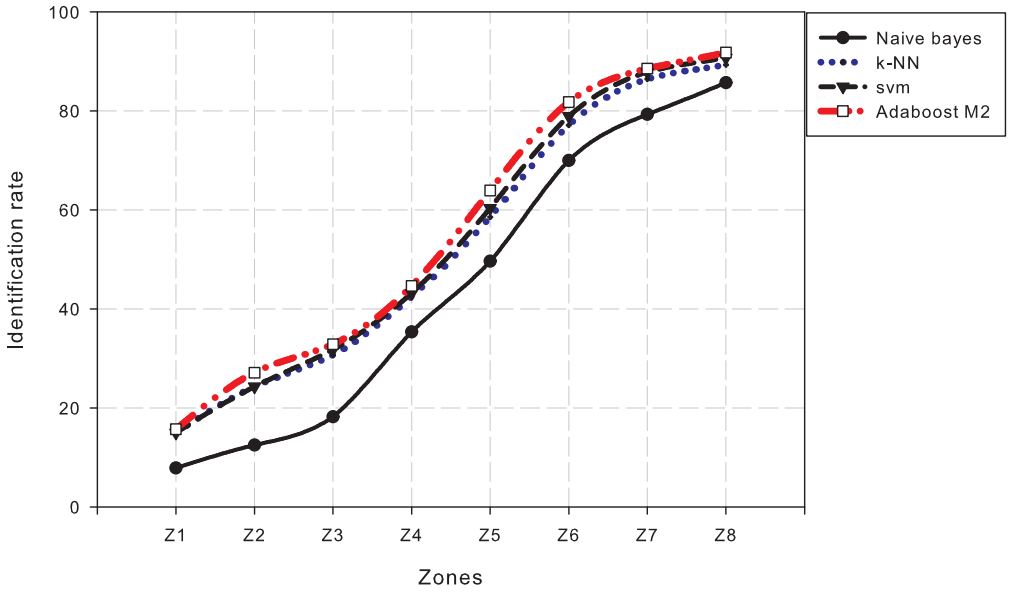


Figure 3.13: Performance of difference classifiers at different zones

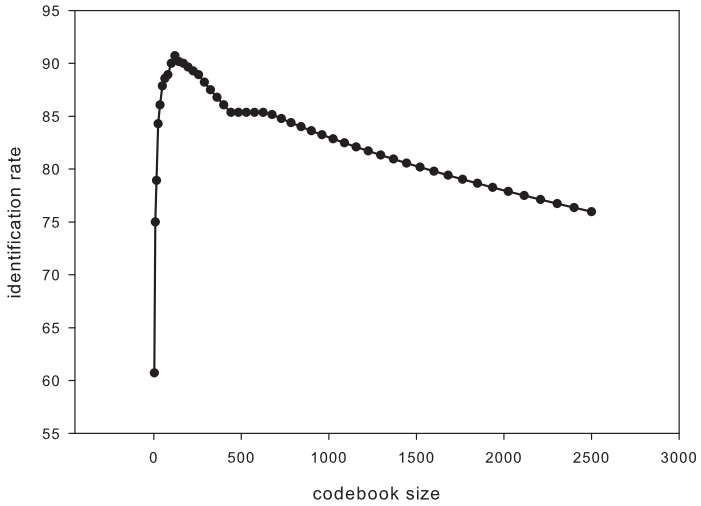


Figure 3.14: Performance across different codebook size

3.6 Summary of the chapter

This chapter discusses a grapheme based writer identification scheme for Malayalam handwritten documents. The scheme has been tested on testbed of280 writers. Experiments were done in detail using different classifiers like Naive-Bayes, k-NN, SVM and Adabboost M2, of which Adabboost and SVM outperformed others. Each grapheme was identified with the features like chain code pair (*gf1*), local stroke direction (*gf2*), curvature (*gf3*) and angle pair (*gf4*). The influence of these features taken individually and in combination on the identification rate was investigated and conclusions were drawn. The investigation of different classifiers at different zones of text led to the conclusion that Adaboost is a stable classifier. Also the impact of codebook size on identification rate was experimented and it was deduced that as the size of codebook increases, identification rate deteriorates beyond 121.

Chapter 4

Writer Identification using Character level features

A writer identification scheme using the salient features of Malayalam characters is described in this chapter. As the success rate of any scheme is highly dependent on the features extracted from the documents. The process of feature selection and extraction is highly relevant and is given more importance. This chapter describes a set of novel features that can be effectively used for Malayalam language. These features are used to form the knowledge vector. This knowledge vector is then used in training as well as the identification phases of the system.

4.1 Introduction

Malayalam scripts are curvaceous in nature with loops and curves. This special nature has been harnessed to determine the characteristics of each stroke of the characters made by different individuals. Almost 85% of characters contain loops in it.

Two prominent ways of writing Malayalam scripts exists today. One followed by the older generation and the other followed by the younger generation. This grouping will be helpful for the first level of clustering of writers. Some people belonging to older generation do rarely exhibit the habit of writing two or three characters connected.

Due to the low allographic variation in Malayalam handwritings, the writer identification scheme needs a prominent feature vector for automatic writer identification. As of that, characters which don't form a loop cannot be so decisive in the feature vector. Also certain characters of a writer do not give an individuality to help for identification. Prominent features can be observed in the hooks made by the users. So dehooking is avoided in the automatic writer identification. Also breakage in loops is taken care by means of dilation.

In Section 4.2 design of the scheme is given. Section 4.3 describes the overview of features. Section 4.4 outlines the implementation details. Section 4.5 analyses the results obtained and provide valid conclusions. The chapter is concluded in Section 4.6.

4.2 Scheme Design

The important phases of the scheme are training phase and identification phase. Fig4.1 depicts the detailed system design of the writer Identification using character level features.

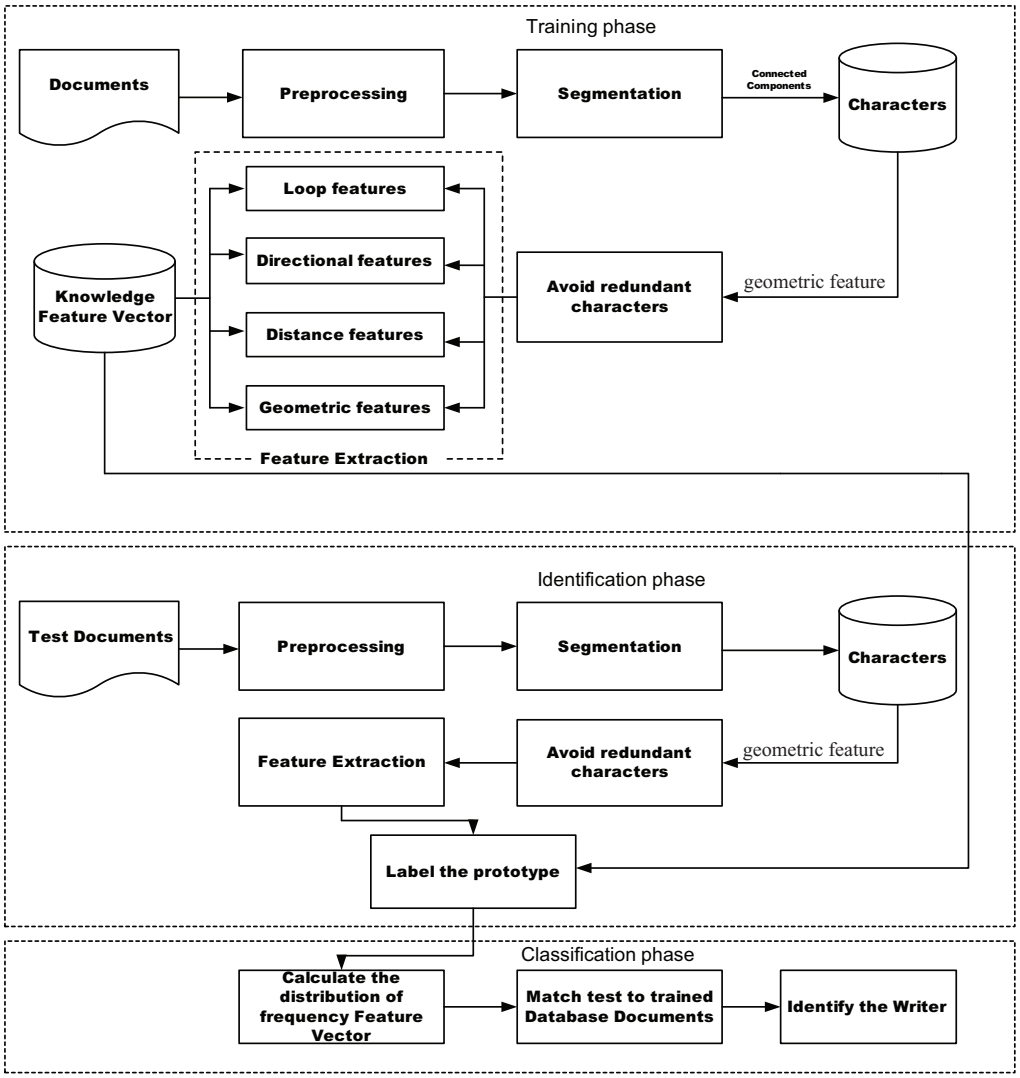


Figure 4.1: System Architecture

In training phase each document of a writer is divided into different zones where each zone comprises of three lines of writing. This division into three lines is obtained from the horizontal projection profile histogram of the binary image obtained through preprocessing. Each Malayalam character is considered as a connected component being determined by segmentation of the different zones using the connected component algorithm. Loop features, directional features, distance features and geometric features are the features obtained from the feature extraction module. Knowledge feature vector would be obtained by feature extraction only after eliminating redundant characters by using the geometric features. On the identification, the modules in the training phase would be repeated as the test documents were preprocessed and segmented to obtain the connected components. A feature vector is obtained by eliminating the redundant characters from the feature extraction module would lead to the classification module to identify the writer.

4.3 Overview of Character level Features

4.3.1 Loop features

Loop features of a stroke include loop including loop area, loop radius and loop roundness [97]. Since most of the Malayalam characters are circular in shape, the loops and curves have to be dealt with minutely. It provides essential information for distinguishing different writing styles. After the study it has been concluded that a writer maintains his own style of loops and curves throughout his writings. The Malayalam character ത(tha) is considered to explain the characteristics of loop features.

Loop roundness (*cf1*)

It is observed that every writer maintains his own shape of roundness for characters with loops throughout his writing. Loop roundness measures its similarity to a perfect circle. The index of dissimilarity, d can be mathematically computed as [39].

$$d = \frac{d^{'}}{C.radius} \tag{4.1}$$

Where $d^{'}$ is computed as:

$$d^{'} = \frac{\sum_{p \in LC} \left(\sqrt{(p.row - C.row)^2 + (p.col - C.col)^2} - C.radius \right)^2}{n(LC)} \tag{4.2}$$

Where $n(LC)$ is the count of the pixels at the edges of loop (loop count, LC) and $C.row$ and $C.col$ are the x & y co-ordinates of the perfect circle and $C.radius$ is the radius of the perfect circle. The circle centre is estimated at the centre of gravity of the loop. It is calculated as

$$C.row = \frac{\sum_{p \in LOOP} p.row}{n(LOOP)} \tag{4.3}$$

$$C.col = \frac{\sum_{p \in LOOP} p.col}{n(LOOP)} \tag{4.4}$$

Where $n(LOOP)$ is the count of pixels that belong to the loop.

Loop Slant (*cf2*)

Each user has a habitual parameter of applying his own style of slant which would repeatedly appear in his characters. Loop slant is measured as the angle of line points of connecting the centre of gravity, as shown in Fig.4.2.

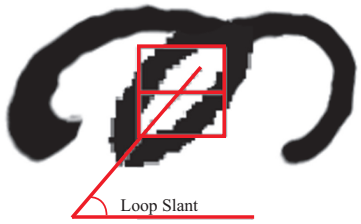


Figure 4.2: Loop slant in the Malayalam letter ത 'tha'

Relative Width/Height ratio of loops (*cf3*)

The maximum and minimum of x & y values of the inside of the loop are calculated and their differences Δx and Δy are computed. The Width/Height ratio is calculated as

$$Ratio = \frac{\Delta x \; of \; the \; loop}{\Delta y \; of \; the \; loop} \tag{4.5}$$

4.3.2 Directional features

Direction angle of the loop (*cf4*)

This is used for distinguishing the broad and narrow loops of the writers. Loops can be ascending or descending. The angle subtended by each point of the loop with the point of intersection is found out as shown in Fig 4.3. The average and the standard deviation of these values are calculated and then the probability density function (PDF) corresponding to each angle values is found out. Obtain the average of these PDF and consider it as the directional angle feature *cf4* of the loop.

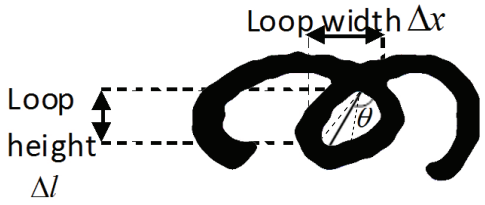


Figure 4.3: Direction angle of the loop in the Malayalam letter ത 'tha'

Direction angle of the character (*cf5*)

It is the average of the angle subtended by the different points of the character with the centroid of the character as shown in the Fig.4.4. An angle subtended by a point of the character with the centroid if the character can be found out with eqn. 4.6.

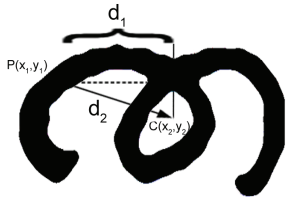


Figure 4.4: Direction angle of the letter ത 'tha' in Malayalam

$$Angle\ (\phi) = \cos^{-1}\frac{d_2}{d_1} \tag{4.6}$$

where $d1$ & $d2$ ... are distances as shown in the Fig.4.4

Slant of a character (*cf6*)

It is the angle of the character formed with its baseline. It is computed using the Algorithm 2 given in [98].

Algorithm 2 Slant of a character

Let say an image denoted by P where $P \in \{0, 1\}$

$$P_{i,j} \in P \left\{ \begin{array}{l} i = 1, 2, ..., h \\ j = 1, 2, ...\text{w} \end{array} \right\}$$

h, w is height and width of P respectively.

1. Take left most $P = \{P_{i,j} | P_{i,j} = 1\}$ define it as $P_1 = (x_1, y_1)$
2. Take first maxima from left most as $P_2 = (x_2, y_2)$
3. set $P_3 = (x_2, y_1)$
4. calculate as distance of $P_1 - P_2$

$$\|d_1\| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$
$$\& \; d_2 \text{ as distance of } P_2 - P_3$$
$$\|d_2\| = \sqrt{(x_3 - x_2)^2 + (y_3 - y_2)^2}$$

5. calculate slope to set shear direction k (clockwise or anticlockwise)

$$m = \frac{y_2 - y_1}{x_2 - x_1}, \; \text{if } m < 0 \text{ then } k = -1 \text{ else } k = 1 :$$

6. calculate slant angle $\theta = k \times \sin^{-1}\left(\frac{d_2}{d_1}\right)$

Curvature of the character (*cf7*)

This feature can be found out using either the point based method or the contour based method. The curvature of the character can be found out by the point based method as described in Algorithm 3

From the experiment results given in Table 4.1, it can be seen that the contour method is superior than the point based method. The details of the contour method as curvature(*gf3*) in section 3.2.5.

Algorithm 3 Point Based Curvature

1. Divide the character into 'n' equal points
2. Calculate curvature at (x_i, y_i) using eqn. 4.7

$$\kappa = \frac{\hat{x}' \cdot \hat{y}'' - \hat{x}'' \cdot \hat{y}'}{\left(\hat{x}'^{2} + \hat{y}'^{2} \right)^{\frac{3}{2}}} \tag{4.7}$$

   where $\hat{x}_i'$ and $\hat{y}_i'$ is the first derivative of (x_i, y_i) and $\hat{x}_i''$ and $\hat{y}_i''$ is the second derivative of (x_i, y_i)

3. Find the average of the κ values of 'n' points and it is the curvature feature (*cf7*) of the character.

4.3.3 Distance features

Distance from the centroid (*cf8*)

It is average of the distance between the points of the stroke and centroid of the character. For example consider the point $P1(x_1, y_1)$ on the stroke let $C(x_2, y_2)$ be the centroid of the character as shown in Fig.4.5. Then the distance between the points P_1 and C can be computed using Eqn. 4.8. If the character is sampled by n points then the average of the distance of these n points from the centroid is computed and it is the feature (*cf8*) of the character.

Then distance between the points $P1$ and C is

$$\text{Distance} = \sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2} \tag{4.8}$$

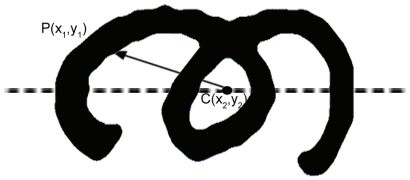


Figure 4.5: Distance feature of the ത 'tha' character in Malayalam documents

4.3.4 Geometrical features

Elliptic features (*cf9*)

Modeling of data using elliptical features gives information regarding how the shapes are written. Stroke, the basic element in the handwriting is composed of three statistical parameters (a, b and θ) [100] as shown in Fig 4.6. Since Malayalam comprises of characters of similar shape, most of them can be modeled by the elliptical trajectory. Thus each stoke is modeled using three parameters a, b and θ . In this method an ellipse is fitted over the character and its component loops. For example, for the character അ 'A' given in Fig. 4.7 '5' ellipses are fitted. Each ellipse can be represented with the three statistical parameters (a, b and θ) where 'a' is the half of the major axis and 'b' is the half of the minor axis and constitutes the angle subtended by the ellipse with the reference axis as shown in Fig 4.6. These parameters characterize the geometric properties of the writing. Essentially the elliptical feature (*cf9*) has 4 components they are average of the 'a' values, average of 'b' values, average of θ values and total number of ellipses fitted over the character.

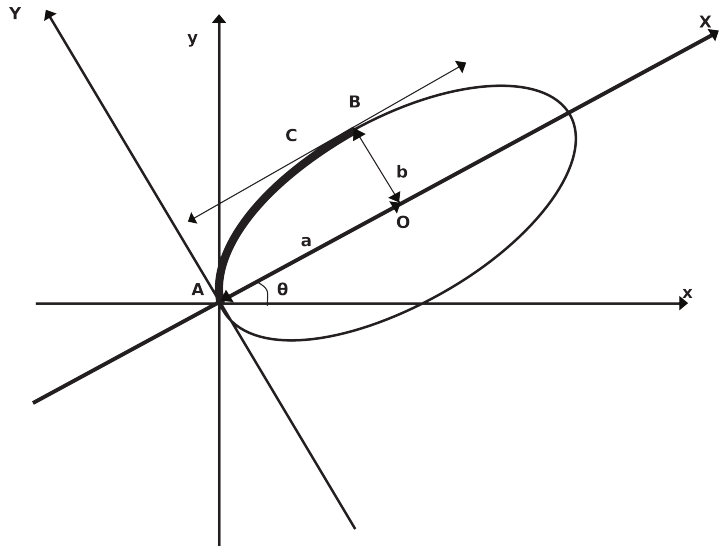


Figure 4.6: Elliptical arc representation

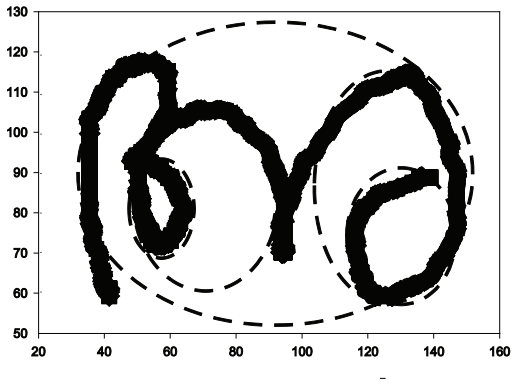


Figure 4.7: Elliptical representation of letter ಅ'A'

4.4 Implementation

The MHDC dataset used for the grapheme based scheme given in chapter 3 was used in this writer identification scheme. Preprocessing and segmentation phases are done using the methods as described in chapter

3. Preprocessing and segmentation phases are done as described in the section 3.2.1 and 3.2.2 respectively. Implementation steps are summarized as described in Algorithm 4.

Algorithm 4 Implementation Algorithm

1. For each document i of the writer w,($d_i{}^w$), divide the document into different zones where each zones is a collection of three consecutive lines.
2. Extract each character C_i in that zones using the connected components algorithm
3. Fit Elliptical curves over each connected component C_i and also on its loops.
4. Obtain the elliptic features ($cf9$)(a, b, θ and number of ellipse) for a each character.
5. Eliminate the redundant characters on the basis of the elliptic features and the chi-square distance measure.
6. Find the feature vector corresponding to the remaining n distinct characters using the features from *cf1* to *cf9* as described in section 4.3.
7. Find the mean and standard deviation and the PDF of each feature.
8. Find the average of the PDF value of each feature and store it as knowledge vector of writer w. A fragment of the knowledge vector is given in Appendix C.

4.5 Experimental observations

Curvature feature ($cf7$) was calculated using the two methods namely point based method or the contour based method. It was observed that that the contour method is superior than the point based method as the number of writers increased. Table 4.1 shows the comparison of point based method and contour based method with respect to writers.

In this study for identifying writers four classifiers namely Naive-Bayes,

Table 4.1: Comparison of point based and contour based curvature feature

Number of writers	Point based curvature feature (%)	Contour based curvature feature (%)
10	100	100
25	100	100
50	98	100
75	96	100
100	90	98
150	84.66	97
200	82.5	96.5
250	80.8	95.6
280	78.571	93.92

k-NN, SVM and Adabboost M2 were tried. A comparison of their performances is given in Fig 4.8. It can be seen that k-NN, SVM and Adabboost M2 showed comparable performance while the performance of the Naive-Bayes decreased as the number of writers increased.

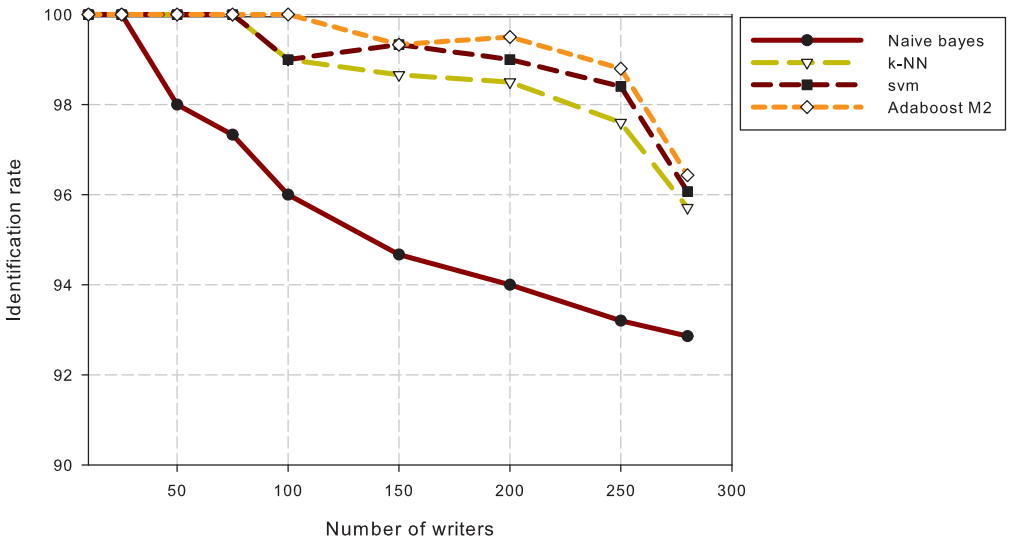


Figure 4.8: performance of the different classifiers

It was also observed that the length of writing (zones) given for identifying the writer has a dependence on the identification rate. The zone for identification can be one word, two word, one line, two lines, three lines, four lines, one paragraph or one complete page. It was observed that the accuracy of identification increased as the length of the zone writing given for identification increased. Fig 4.9 gives a comparison of this fact among different classifiers. It can be seen that the zone with maximum amount of text yielded greater identification rate across the classifiers.

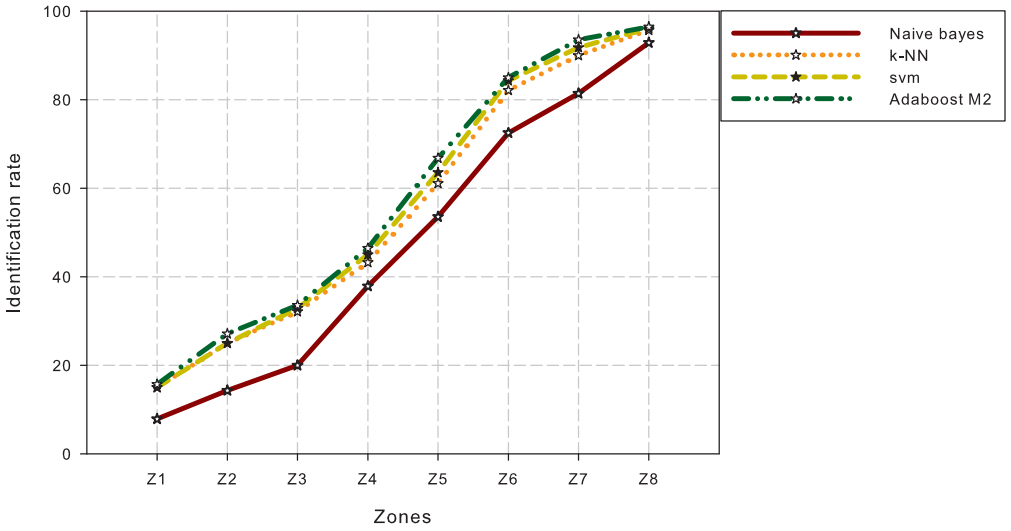


Figure 4.9: performance with respect to amount of text

Even though nine features were identified for describing each writer a study was conducted to find the influence of each feature on identification rate of the model. Table 4.2 gives comparison of these features when considered as alone and taken together. It can be seen that the performance of the model improved very much when all the nine features are considered together.

Table 4.2: Comparative evaluation of features

Number of writers	***cf1***	***cf2***	***cf3***	***cf4***	***cf5***	***cf6***	***cf7***	***cf8***	***cf9***	**Combined feature (cf1+cf2+cf3+ cf4+cf5+cf6+ cf7+cf8+cf9)**
10	80	70	90	80	80	60	100	70	100	100
25	80	68	88	80	80	56	100	64	100	100
50	78	66	88	78	80	56	100	64	96	100
75	77.33	65.33	88	78	78.66	54.66	100	64	94.66	100
100	77	63	87	76	77	54	98	62	93	100
150	76	62	86.66	74.66	75.33	52.66	97	60.66	92.67	99.33
200	75	62	86	74	75	51	96.5	59	92	99.5
250	74	61.2	84.8	72.8	73.6	49.2	95.6	57.2	91.6	98.8
280	72.85	59.64	82.5	70.71	71.78	47.5	93.92	55.35	91.07	96.43

4.6 Summary of the chapter

This chapter describes a set of character level features that can be used for the writer identification. They include loop features, directional features, distance features and geometric features. Classification algorithms like Naive-Bayes, k-NN, SVM and Adabboost M2 were considered. It was found that Adabboost M2 outperformed other classifiers as the number of writers increased.

Chapter 5

Writer Identification using image processing techniques

In this chapter writer identification is tried using two image processing techniques namely Wavelet Domain Local Binary Patterns (WD-LBP) and Scale Invariant Features Transform (SIFT). The schemes are tested on a test bed of 280 writers and their performances are evaluated. It was found that SIFT feature outperform WD-LBP but the time complexity associated with SIFT feature is high as compared with that of WD-LBP.

5.1 Introduction

The identification rate of a writer identification scheme is highly dependent on the feature selection and feature extraction phases. In this chapter, two feature selection and extraction methodologies using the image processing technique, namely, Wavelet Domain Local Binary Patterns (WD-LBP) and Scale Invariant Features Transform (SIFT) are explained.

The style of writing scripts vary in different writers with the curvature and slant they use in it. As these cannot be revealed easily through traditional texture characteristics, wavelet domain is utilized to achieve this. Hence WD-LBP was used which could reveal the intrinsic features of the writing scripts. Here the features are extracted according to the global parameters of wavelet coefficients distribution.

SIFT features have been extensively utilized in pattern recognition and classification mostly in object recognition. There is a tradition of the usage of SIFT features in robust digital watermarking [105], face authentication [106], graffiti tags recognition [107], car make and model recognition [108] and fingerprint verification [109]. Local features based on SIFT are somewhat invariant to many of the sources of variability. Locality is important, because even if occlusions obscure some portions of the handwriting, there may be enough local features extracted elsewhere which will help to classify the image correctly. Hence the SIFT technique was also tried for writer identification.

In this chapter, Section 5.2 details the system architecture of the writer identifications scheme for Malayalam language. Section 5.3 describes the feature extraction techniques of WD-LBP and SIFT. Section 5.4 explains the codebook generation. Section 5.5 provides the implementation details. Result analysis is done in Section 5.6. Chapter is concluded in Section 5.7.

5.2 Scheme Design

The system architecture of writer identification scheme in document level is illustrated in Fig. 5.1.

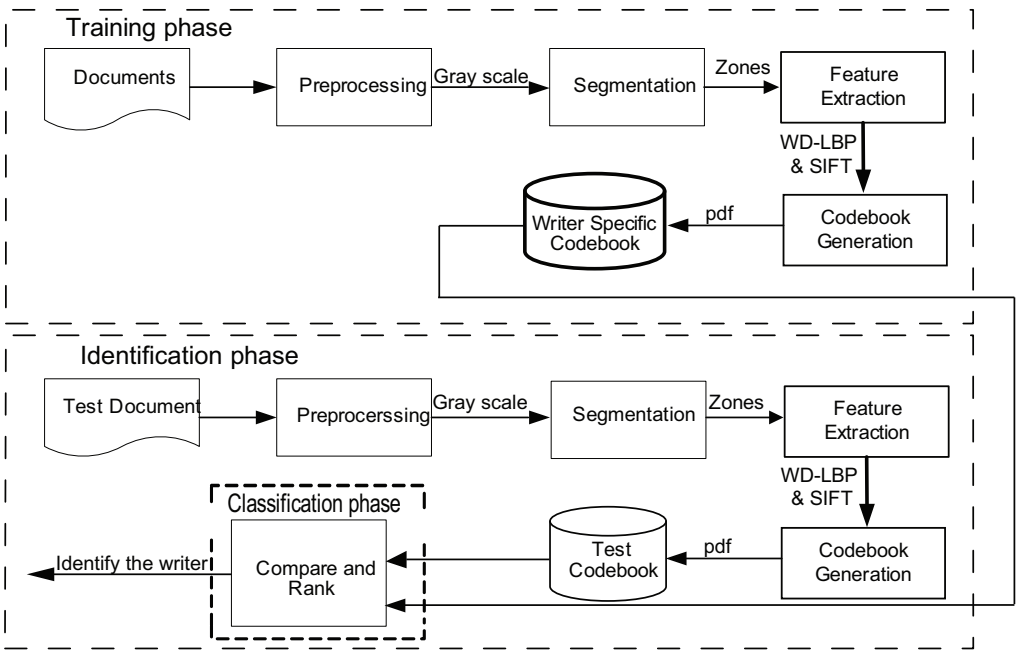


Figure 5.1: System Architecture

5.2.1 Preprocessing

The main purpose of this is to remove unwanted areas and noise of the raw input image as it will affect the processing in subsequent phases. The naive image of documents should be pre-processed at first hand through Algorithm 5.

5.2.2 Segmentation

In this module the whole document of writer is divided into different zones to capture their individuality. The total number of zones is 4 for the

Algorithm 5 Preprocessing

1. If the technique is SIFT
   (a) Calculate the mean intensity and standard deviation of the gray scale image
   (b) Find out a threshold value based on the mean and standard deviation such that points above that value can be classified as white and vice versa.
   (c) Convert the gray image into binary image around the above threshold value.
   (d) Image de-noising is practiced to attain the perfect binary image of the documents [72][89].
2. Else if the technique is WD-LBP
   (a) Convert the scanned document into 8-bit gray scale image

extraction of WD-LBP feature where as the total number of zones is set to be 6 for SIFT feature.

5.3 Overview of Features

This phase is the essence of the system. Here we incorporate different techniques and schemes to extract the individuality features attributed to a writer.

5.3.1 Wavelet Domain Local Binary Patterns (WD-LBP)

This feature is the local structure information in wavelet domain texture described by LBP where wavelet domain texture patterns are composed by wavelet coefficients [110]. WD- LBP feature(*fd1*) is obtained by the local binary pattern operations upon the s-level wavelet decomposition of

the training samples of the handwritten documents of the writers. It is calculated as follows.

1. Obtain the gray level image of the documents
2. Divide the document into 4 zones
3. Decompose each documents into s-level using the wavelet transform function $Wf\ (x,\ y) = I(x,\ y)\ *\ \psi(x,\ y)$ where, '*' stands for the two-dimensional convolution operator, and $\psi(x,\ y)$ is a two-dimensional wavelet function and $I(x,\ y)$ is the gray function which indicates the gray level at $(x,\ y)$ pixel.
4. LBP codes of these sub bands $Ws,\psi(x,\ y)$ is calculated by using LBP operator by considering the difference between the gray value of the pixel X and its symmetric neighbor set of P pixels. LBP value for the center pixel $(x,\ y)$ of image I(x,y) can be obtained through:

$$LBP_{P,R(x,y)} = \sum_{p=0}^{P-1} s\left(I\left(x,y\right) - f\left(x_p,y_p\right)\right) 2^p \tag{5.1}$$

P represents the number of the circularly symmetric neighborhood and R is the radius of the neighborhood. When a neighbor does not fall exactly in the center of a pixel, its value is obtained by interpolation. $s(z)$ is the thresholding function:

$$s\left(z\right) = \begin{cases} 1, z \geq 0 \\ 0, z \langle 0 \end{cases} \tag{5.2}$$

By using the center pixel as a threshold, a binary pattern can be produced. Then, a LBP code is produced by multiplying the binary pattern with weights assigned to the corresponding pixels, and summing up the result. Different local binary patterns point

to different local structures. Here, a moving window of LBP is moved along the wavelet coefficients in each sub bands. Also the absolute value of coefficients for LBP code is used because high frequency sub bands contain both positive and negative coefficients. The feature vector has 2^P elements. P is the number of sampling points. A value of $R = 1$ and $P = 8$ indicates the 8-neighbor positions of the center.

5. The LBP code of each wavelet sub band, is then found as follows

$$LBP^{p,r}_{s,\Psi}(m,n) = \mathrm{LBP}^{p,r}\left(W_{s,\Psi}(m,n)\right) \tag{5.3}$$

6. The LBP histogram is then computed as follows

$$\begin{aligned} &H^{p,r}_{s,\Psi d} = [l_1\, l_2 \ldots\ldots l_k \ldots. l_{2^p}] \\ &\text{where } l_k \text{ denotes the number of patterns and k} \in \ 1,\ 2,\ 3, ...,\ 2^{\mathrm{p}} \end{aligned} \tag{5.4}$$

$$\begin{aligned} &l_k = \textstyle\sum_{m,n} \delta\left\{LBP^{p,r}_{s,\Psi d}(m,n),\ k\right\}, \\ &where\ \delta\{i,j\} = \left\{{}^{1,\ i=j}_{0,\ i\neq j}\right\}\ \ and\ \ \ d \in \{0,1,2,3\} \end{aligned} \tag{5.5}$$

denotes the index of wavelet sub bands.

7. A final histogram is obtained by concatenating all the sub band histograms. The final histogram constitutes the C_i of the i^{th} writer, which can be depicted as follows.

$$C^i = \left[H^{p,r}_{S,\Psi 0},\ H^{p,r}_{S,\Psi 1}, \ldots\ldots\ldots\ldots, H^{p,r}_{1,\Psi 3}\right] \tag{5.6}$$

Thus the feature vectors corresponding to all writers are stored in the form of C_i.

5.3.2 Scale Invariant Features Transform (SIFT)

SIFT (Scale Invariant Feature Transform)(*fd2*) is used to extract distinctive invariant local features from images [111].That is, the SIFT feature algorithm is based upon finding locations within the scale space of an image which can be extracted for analytical purpose. The relevance of the algorithm is that it generates a large number of features on a wide spectrum of scales and location. The number of features generated is directly related to the image size, content and algorithm parameters. Detecting Features invariant to the fluctuations in glow, image noise, rotation, scaling etc are in four stages:

- Scale- space Extrema detection
- Keypoint localization
- Orientation Assignment
- Generation of Keypoint descriptors

Stage 1: Scale-space Extrema detection

The first stage namely scale space extrema detection, searches over scale space using a difference of Gaussian function to identify potential interest point those are invariant to scale and orientation. Adjacent Gaussian images are subtracted to produce difference of Gaussian images are shown in Fig.5.2 where the key points are identified as local maxima and minima of difference of Gaussian in scale space. The scale and location features of a document is defined as the function $L(x, y, \alpha)$,which is convulsion of a variable scale Gaussian $G(x, y, \alpha)$ with an input document image $I(x, y)$ as follows[111]

$$L\left(x, y, \alpha\right) = G\left(x, y, \alpha\right) * I\left(x, y\right) \tag{5.7}$$

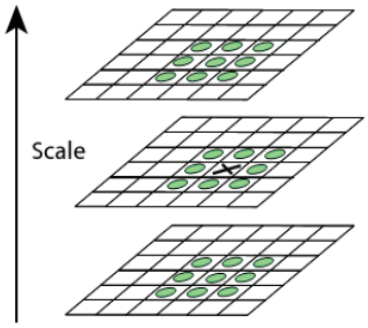


Figure 5.2: Scale space extrema detection (Reproduced from [111])

where * is the convolution operator in the x and y directions and

$$G(x, y, \alpha) = \frac{1}{(2\pi\alpha^2)^{\frac{1}{2}}} \exp\left(-\frac{x^2 + y^2}{2\alpha^2}\right) \tag{5.8}$$

The keypoints those are invariant to scale and orientation is provided by the difference of Gaussian images across the scales. The difference between two nearby scales, D(x,y,α) separated by a constant multiplicative factor

k is given by

$$\begin{aligned} D(x, y, \alpha) &= (G(x, y, k\alpha) - G(x, y, \alpha)) * I(x, y) \\ &= L(x, y, k\alpha) - L(x, y, \alpha) \end{aligned} \tag{5.9}$$

The keypoints are identified as local maxima and minima of the DoG writing images across scale. Each pixel in the DoG is compared to other 8 neighboring pixels at the same scale and 9 corresponding neighbors at the neighboring scales. If the keypoint is the local maxima or minima, it is selected as a candidate keypoint.

Stage 2: Keypoint localization

The second stage determines the location and scale of each candidate keypoint where measures of stability and interpolation of nearly data are well considered.

Stage 3: Orientation Assignment

The Third stage computes a gradient orientation histogram in the neighborhood of the key point .One or more orientations are assigned to each keypoint based on local image gradiants. Any keypoint that is within 80% of the highest peak is used to create a separate keypoint.The Orientation assignment of each key point is obtained by computing the gradient magnitude M(x,y) and orientation θ(x,y) of the scale space for the scale of that keypoint:

$$M(x,y) = \sqrt{(K(x+1,\, y) - K(x-1,\, y))^2 - (K(x,\, y+1) - K(x,\, y-1))^2} \tag{5.10}$$

$$\theta(x,y) = \arctan \frac{K(x,\, y+1) - K(x,\, y-1)}{K(x+1,\, y) - K(x-1,\, y)} \tag{5.11}$$

All the properties of the key point are measured relative to the keypoint orientation. This caters for rotation invariance.

Stage 4: Generation of Keypoint descriptor

The Fourth stage measures local image gradient at the selected scale in the region around each keypoint. Feature descriptors are computed as a set of orientation histograms on 4 x4 pixel neighborhood. The SIFT feature with 4x4x8=128 values is given by the gradient magnitude and by a Gaussian

which has then to be normalized. Among the shape descriptors, SIFT features are more prominent in pattern recognition and classification as they come with

- **Locality-Features** detected are local and robust to clutter and occlusion.
- **Distinctiveness** Individual features can be matched to a large database.
- **Quantity** Many features can be generated even for small objects.
- **Efficiency** for real time performance.
- **Extensibility** They can be extended to different dimensions each with added robustness.

5.4 Code book generation

The codebook is generated according to the Algorithm 6 and 7.

5.5 Implementation

The MHDC dataset used for the grapheme based scheme given in chapter 3 was used in this writer identification scheme. WD-LBP and SIFT features of the document in the training set were extracted and the codebook for the writers were created.
In the identification phase, a query descriptor (test document) pass through the modules in training phase where the features are extracted and a feature vector corresponding to the query descriptor could be obtained. This would be classified with writer specific codebook of trained documents. Prepare a sorted hit list with increasing distance value (similarity score) between the query descriptor and writer specific codebook. Select the first ranked sample which will ideally identify the writer.

Algorithm 6 CodeBook Generation using WD-LBP

1. Obtain the gray level image of the documents
2. Divide the document into 4 zones.
3. Decompose each zone into s-level using the wavelet transform function $Wf(x,y) = I(x,y) * \psi(x,y)$ where, '*' stands for the two-dimensional convolution operator, and $\psi(x,y)$ is a two-dimensional wavelet function and $I(x,y)$, gray function which indicates the gray level at (x,y) pixel.
4. LBP codes of these sub bands $Ws, \psi(x,y)$ is calculated by using LBP operator by considering the difference between the gray value of the pixel X and its symmetric neighbor set of P pixels.
5. The LBP patterns constitute a feature vector which is called LBP histogram for one zone by taking the probability density function (PDF) of each sub bands and its average.
6. Obtain the writer specific feature vector by taking the average of all LBP histograms and store it as a writer specific codebook entry.

Algorithm 7 CodeBook Generation using SIFT

1. Split each document into equal sized zones. Total no. of zones is set to be 6 for SIFT feature.
2. Extract key point descriptors from each zone.
3. Calculate the probability density function (PDF) of each descriptor over the zones to obtain a fused feature vector of dimension 128, which is a representative of a zone.

5.6 Experimental results

A comparison of the efficiency in identifying writers was measured with respect to the number of writers and the observations are given in Table 5.1. It was observed that the time complexity associated with the SIFT feature was higher than WD-LBP. When a page consisting of 21 lines with 30 characters per line was used for comparing the WD-LBP and SIFT techniques, it was found that WD-LBP took only 2.8 seconds for feature extraction where as SIFT took 12.6s.
The classification process is evaluated using different classifiers like Naive-Bayes, k-NN, SVM and Adaboost M2. Efficiency was measured using identification rate where, identification rate of this system was calculated by choosing the successfully identified writers without any false positive.
As shown in the following Table 5.1, as the number of writers' increased, the performance of Naive-Bayes when using WD-LBP feature extraction method deteriorated faster than the usage of SIFT features. k-NN, SVM and Adaboost classifiers show similar behavior with better performance in SIFT features.

Different wavelets like db4 and Haar were used for the purpose of decomposition while using WD-LBP features. The performance of the classifier when using both of them is plotted in the Fig 5.3. It is evident that the db4 wavelet gives good results when compared with Haar. Studies have shown that higher order wavelets give better performance for curvaceous characters. Since Malayalam scripts is abundant with loops and curves db4 outperformed Haar which is a low order wavelet.

Also in computing SIFT features, the variability between sample and codebook distributions, is computed in terms of chi-square distance and Euclidean distance as well. Fig5.4 shows that Chi-square distance outperforms Euclidean distance measure as the number of writers increases.

Table 5.1: Performance based on WD-LBP and SIFT features

Identification rate with naive bayes classifier

Number of writers	WD-LBP	SIFT
10	90	100
25	88	96
50	86	92
75	84	90.67
100	79	88
150	74	84.67
200	72	82
250	67.6	79.6
280	66.07	78.21

Identification rate with k-NN classifier

Number of writers	WD-LBP	SIFT
10	100	100
25	92	96
50	86	94
75	85.33	92
100	81	90
150	78	89.33
200	75.5	86.5
250	71.6	84.8
280	70.36	83.57

Identification rate with SVM classifier

Number of writers	WD-LBP	SIFT
10	100	100
25	92	96
50	86	94
75	85.33	93.33
100	83	92
150	79.33	90
200	76.5	87
250	74	85.2
280	71.07	84.29

Identification rate with Adaboost M2 classifier

Number of writers	WD-LBP	SIFT
10	100	100
25	96	100
50	92	96
75	88	94.67
100	85	93
150	80.67	90.67
200	77	88
250	74.8	86.8
280	73.21	85.714

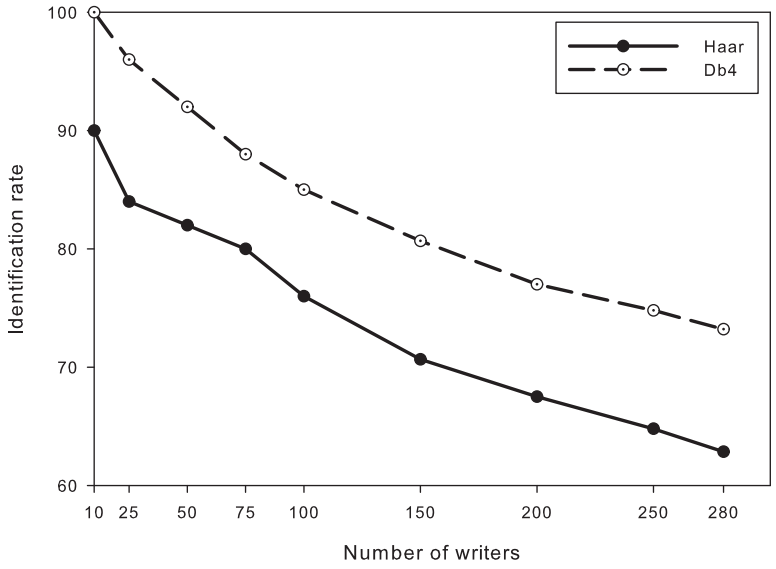


Figure 5.3: Comparative results of different wavelets used for decomposition

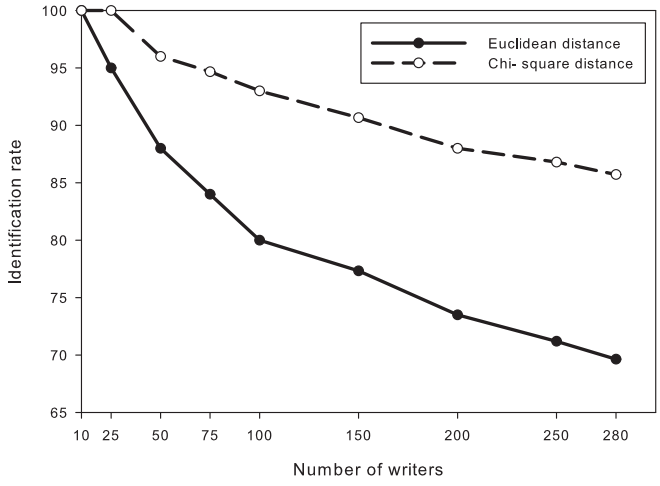


Figure 5.4: Comparative results of different distances used for SIFT features

Stability test is conducted to check the stability of the features and classifiers in writer identification. This is done by analyzing the performance of features at different reference points /zones (one word - two words; one line - two lines - three lines - four lines; one paragraph; full page.) on the documents of the total 280 writers. The handwritings are divided in terms of zones as depicted in the Fig. 5.5. and Fig.5.6. The purpose of such a division is to detect the influence of the amount of texts on the identification rate. It is seen that the zone with maximum amount of text yielded greater identification rates as compared with zones having less text for different classifiers.

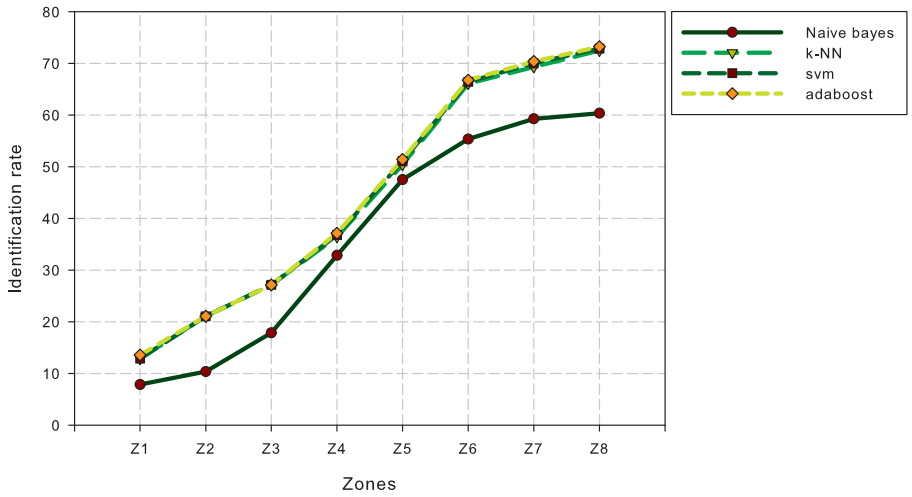


Figure 5.5: Performance of different classifiers on WD-LBP feature with respect to amount of text

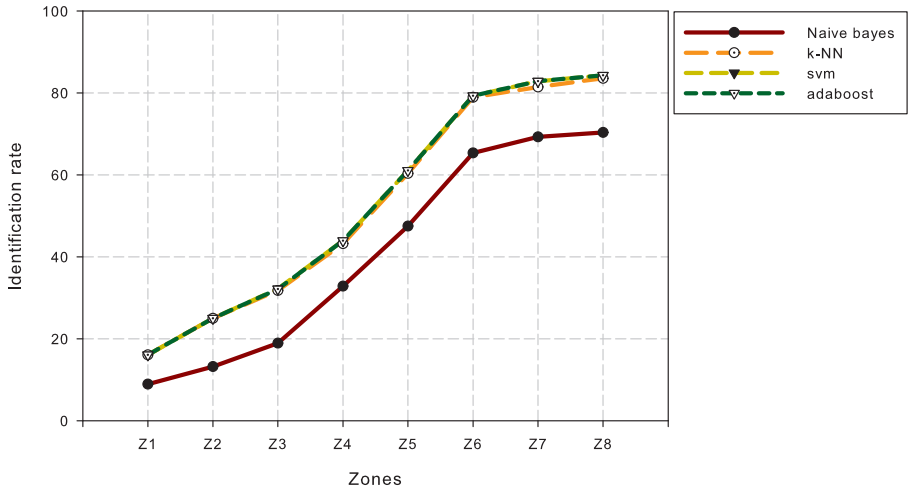


Figure 5.6: Performance of different classifiers on SIFT feature with respect to amount of text

5.7 Summary of the chapter

Image processing techniques like WD-LBP and SIFT were used for writer identification of which SIFT gave better identification rate at the expense of high time complexity.

Chapter 6

Result analysis and discussions

Three schemes for writer identification of Malayalam documents is outlined with sufficient substantiations in earlier chapters. Thorough discussions of the results drawn in the earlier chapters are done here. Inferences/conclusions are drawn/ reached from which the next milestones are identified. This paved way for the schemes like online character recognition and historic document analysis.

6.1 Introduction

The selection of an appropriate feature vector is imperative in determining the success rate of a writer identification scheme. All the phases - training and identification should be taken into account separately and generally. Consideration should be given to the language selected, its characteristics, history and evolution, peculiar nature in presentation, developments in literature etc. along with the possible fluctuations in individual character style, changes occurred in the periodical evolution etc.

The scintillating significance of writer identification is ever increasing as technology is having tremendous growth. Every piece of writing, twaddle to scholastic, keeps individuality in its variants leaving an identity with an analytical importance. Apparently, the slant and roundness were considered to determine the characteristics of individual or personal writing style. Recent developments in character recognition assume allographs to find the individuality of a character. This proposes that each piece of writings would reflect the personality of the author of it. Fragmented connected components can be extracted from the typical character shapes of the imprints that may be stored to compare with scanned handwriting document. The writing direction and curvature of these fraglets can be characterised by their contour.

This chapter deals with a comparison of the performance analysis of the study at grapheme, character and document level which was described in chapters 3 to 5.

6.2 Mathematical Model for Writer Identification Scheme

To identify a person's handwriting, the biggest essentiality is identifying the individuality of the same. It constitutes shape primals, relationships

between different parts of handwriting etc which can be in general termed as distinct primals. The primals can be frequently identified in handwriting. The first step in any writer identification scheme is to extract the primals from the handwritings efficiently and effectively represent them as a feature vector. This is highly important because the overall identification rate of the system is influenced by the features extracted. A proper similarity measure has to be adopted for the purpose of comparison of the feature vectors, thus formulating the second step. Of course, supporting steps needed to be designed by considering the features of the language under consideration. The final step involves a classifier which classifies the handwriting under question to the most likely one with the help of similarity score. The problem of writer identification can be formally depicted as follows.

> An input feature vector T_q is given to determine the writer w_i, where $i \in 1, 2, 3 \ldots, N$

The solution is that the feature vector T_q is compared with all the feature vector samples of the writer present in the database archive using techniques of similarity measurement and arriving at the conclusion that the maximum similarity score point towards the owner of the questioned feature vector. Mathematically, it can be written as

$$Tq \in \left\{ \begin{array}{ll} W_k & \text{if } (\mathrm{k} = \max_{\mathrm{k}} S\left(T_q, T_{wk}\right) \quad \rangle \eta) \\ W_{k+1} & \text{otherwise} \end{array} \right\}, \tag{6.1}$$

where $S(.,.)$ is the similarity function between two feature vectors.

Using the above mentioned concepts, a mathematical model is formulated as follows.

Let $D = \{d_1^w, d_2^w, d_3^w, \ldots d_n^w\}$be the documents from the n writers.

And each $d_1^w = \{d_1, d_2, d_3, \ldots d_t\}$

For each d_i^w, a feature vector is computed as $Fv_i^w = \{f_1, f_2, f_3.....f_s\}$by analyzing $d_j$where $j = 1, 2...t$ and $f_i$is the feature value corresponding to the i^{th} primal. For identification phase feature vector $Fv_t^Q$of the test query document $Q_t$is calculated and $W_i$is the expected writer of the document if $Sim\left(Fv_i^w, Fv_t^Q\right) \rangle Sim\left(Fv_j^w, Fv_t^Q\right)$ for 1 $\leq$ j $\leq$ n ; j $\neq$ i. Also the probability that the given document belongs to a writer can now be computed using Bayesian rule as

$$P\left(W_i/Q_t\right) = \frac{P\left(Q_t/W_i\right) * P\left(W_i\right)}{P\left(Q_t\right)} \tag{6.2}$$

Where

$$P\left(Q_t/W_i\right) = \prod_{j=1}^{s} p\left(f_j^t = f_j^{w_i}\right) or \prod_{f_i \in D} p\left(f_i/W_i\right) \tag{6.3}$$

Using the law of total probability, we can write

$$P\left(Q_t\right) = \sum_{i=1}^{n} P\left(Q_t/W_i\right) * P\left(W_i\right) \tag{6.4}$$

Now eqn 6.2 becomes 6.5

$$P\left(W_i/Q_t\right) = \frac{P\left(Q_t/W_i\right) * P\left(W_i\right)}{\sum_{i=1}^{n} P\left(Q_t/W_i\right) * P\left(W_i\right)} \tag{6.5}$$

6.3 Result analysis and Discussions

The various schemes for writer identification of Malayalam documents-grapheme level, character based and WD-LBP&SIFT based-discussed in the previous chapters and are compared here. Angle pair, curvature and directional features (chain code pair directions and local stroke

direction) were considered in the grapheme scheme. Loop features, directional features, distance features and elliptic features were considered for character level. And for document level, WD-LBP and SIFT features were taken into account. Experiments were conducted to find the effectiveness of each feature and their combinations on the performance of different schemes. It was found that all the features taken together outperformed the effect of individual features on the performance of each of these schemes. This may be due to the characteristic features of the Malayalam script and the low allographic variations among writers. The following four steps are in common in the comparative study of each scheme of writer identification.

- Influence of features in the elimination of redundant characters.
- Stability test on each feature.
- Consistency among features.
- Overall performance analysis.

6.3.1 Influence of features in the elimination of redundant characters

The elimination of redundant characters is required before feature extraction in different schemes as mentioned earlier. However, this is not required when the document is considered as a whole where image processing techniques like WD-LBP and SIFT features are used. A writer's behavior has to be obtained by the maximum number of different characters coming across the given document. The ambiguity has to be avoided by reducing the redundant characters. Three techniques were used for the elimination of redundant characters like randomized selection, height-width ratio & curvature and geometrical feature. This study is done with different number of writers varying from 10 to 280.

Regression Analysis was done to check how the three features (randomized selection, height-width ratio & curvature and geometrical feature) behaved with respect to the number of writers. Regression was applied both grapheme based and character based schemes. $score(y)$ was regressed over x with the Eqn. 6.6

$$y_i = A_i x + B_i + \varepsilon, \quad i = 1, 2, 3 \tag{6.6}$$

Where y_i denotes the score value for the i^{th} technique and i can be technique either randomized selection or height-width ratio & curvature or geometrical features. We have got significant R^2 value for every y_i and it was 0.646727594 for the technique random selection, 0.653652487 for the technique heightwidth ratio & curvature and 0.695097757 for the technique geometrical features.
From the ANoVA table (Table 6.1) it is clear that the regression is significant for all the three techniques. The geometric feature play the most significant role in the elimination of redundant characters.

The parameter estimation of the three different techniques is given in Table 6.2. This table also shows that the technique based on geometrical features provided highest value for the coefficients and lowest value for the standard error.

6.3.2 Stability test of features in each scheme

This test was conducted to check the stability of the features in each scheme for writer identification. This was done by analyzing the performance of features at different reference points (zones as described in section 3.4.1) on the documents of the total 280 writers. Stability of all features in different scheme is described in Table 6.3.

Fig 3.13, Fig 4.9, Fig 5.5 and Fig 5.6 depict the performance of different classifiers at different zones for the schemes *viz.* grapheme, character,

Table 6.1: ANOVA table of different features

ANOVA Y1 (randomized selection)					
	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	345.977	345.977	21.96812	0.000526
Residual	12	188.9886	15.74905		
Total	13	534.9655			
ANOVA Y2 (height-width ratio and curvature selection)					
	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	117.6424	117.6424	22.64728	0.000465
Residual	12	62.33457	5.194547		
Total	13	179.9769			
ANOVA Y3 (geometrical feature selection)					
	df	*SS*	*MS*	*F*	*Significance F*
Regression	1	92.12558	92.12558	27.35688	0.000211
Residual	12	40.41057	3.367547		
Total	13	132.5362			

Table 6.2: Parameter estimation of different techniques

parameter estimates of Randomized Selection			parameter estimates of height-width ratio & curvature			parameter estimates of geometric feature		
	Coefficients	Standard Error		Coefficients	Standard Error		Coefficients	Standard Error
$\hat{A_1}$	92.96074	1.685068	$\hat{A_2}$	96.62299	0.967752	$\hat{A_3}$	98.1819	0.779197
$\hat{B_1}$	-0.06008	0.012819	$\hat{B_2}$	-0.03504	0.007362	$\hat{B_3}$	-0.031	0.005928

WD-LBP and SIFT. It was observed that performance of the Naive Bayes classifier is low when compared to other classifiers. k-NN, SVM and Adaboost M2 classifiers in the document level presents better performance up to Z6 when compared with the same classifiers in grapheme level. But the Naive bayes gives higher performance only up to Z3 when compared with the same classifiers in grapheme level. From these we can make the inference that the document level features are acceptable for small reference points (zones) like maximum up to four lines of data for the k-NN and SVM and up to one line data for Naive bayes.

6.3.3 Consistency among features

Evaluation of the influence of the consistency of the features on identification rate across the number of writers is given in Table 6.4. The coefficient of variation C_v denotes the consistency among features. And it calculated by the eqn 6.7.

$$C_v = \frac{\sigma}{\mu} \tag{6.7}$$

Where σ is the standard deviation and μ is the mean of the features. Consistency is maximum when C_v is minimum. From the Table 6.4 it can be observed that *cf7* is most consistent among 15 features under consideration.

6.3.4 Performance evaluation of classifiers across the three schemes

The measures commonly used for evaluating the performance of classification models are sensitivity, specificity, precision and F-measure. Table 6.5 shows the performance of the classification models, Naive Bayes, k-NN, SVM and Adaboost M2 across the three schemes under study for writer identification. From Table 6.5 it is evident that character based approach is most suitable for writer identification.

6.3.5 Decisive features for Malayalam characters for writer identification

The most decisive features of Malayalam script that can be used for writer identification is worthwhile. In this study for this purpose the identification rate and consistency value of various features across the three identification schemes(grapheme based,character based and image processing based schemes) were considered. Table 6.6 gives a comparison of the features across the three schemes namely grapheme based, character based and document based. For a feature to be decisive the identification rate should be fairly high and the value of coefficient of variation should be low. The experiments had revealed that the curvature and directional features with geometric properties of a character are the most decisive features for writer identification of Malayalam documents. Geometric features such as a,b,θ globally reflect the geometric properties of the set of muscles and joints used in a particular handwriting movement. Thus it has strong relationship with directional features.

6.4 Inferences

In the light of the above discussions, my part of research work in writer identification scheme for Malayalam asserts character level feature is more elegant than the grapheme level or document level. The challenge posed by the writer identification scheme in Malayalam is highly attributed to the

i. Two prominent ways of writing Malayalam scripts (old and new) and

ii. The lack of availability of all conjunct letters from different writers makes the formation of the training set a tedious task for writer identification in offline mode and hence also the same applies for offline Malayalam character recognition schemes.

When the above mentioned attributes is taken into account while building the feature vector, it is noted that there is a considerable increase in the feature space. These factors motivated the need of developing a framework for online character recognition.

Care has been taken in this scheme to generate all the conjunct characters from the 64 basic characters. Obviously online character recognition takes less recognition time than offline character recognition. The tedious task of using English keyboard to enter Malayalam characters is eliminated with the advent of online character recognition.

To accelerate computing mechanisms in the Malayalam regional language, online character recognition provides strong basis. It helps a person to gather information in their own language. Also this type of frame work leads to a component in the multimodal interface for gathering information using the question answering system in regional language. This online character recognition system can be used in other scenarios like online historic document analysis, online writer identification and online signature verification.

With the above distinctive features of Malayalam script, a frame work for online Malayalam character recognition is found feasible. Thus we have developed a frame work for the same and various features were analyzed across different classifiers for online handwritten character recognition. Later this framework has been extended for analyzing the historic documents in Grantha script by recognizing the Grantha script and converts it into the two ways of Malayalam scripts. The detailed description of the development and experimental analysis of these systems will be discussed in the next chapter.

Table 6.3: stability of all features at different level

		Z1	Z2	Z3	Z4	Z5	Z6	Z7	Z8
Grapheme level	fg1	12.14286	17.85714	20.71429	28.21429	40.71429	51.78571	56.07143	58.21429
	fg2	13.21429	19.28571	22.5	30.71429	43.92857	56.42857	61.07143	63.21429
	fg3	16.78571	25.35714	29.28571	40	57.14286	73.21429	79.28571	82.14286
	fg4	17.5	26.07143	30	41.07143	58.57143	75	81.42857	84.28571
	fg-combined	18.92857	28.21429	32.85714	44.64286	63.92857	81.78571	88.57143	91.78571
Character level	*cf1*	14.28571	21.07143	25.35714	35	50.35714	64.28571	70.71429	72.85714
	cf2	11.78571	17.14286	20.71429	28.57143	41.42857	52.5	57.85714	59.64286
	cf3	16.07143	23.92857	28.57143	39.64286	57.14286	72.85714	80	82.5
	cf4	13.92857	20.35714	24.64286	33.92857	48.92857	62.5	68.57143	70.71429
	cf5	18.21429	26.78571	32.14286	44.28571	63.92857	81.07143	89.28571	92.14286
	cf6	9.285714	13.57143	16.42857	22.85714	32.85714	41.78571	46.07143	47.5
	cf7	18.57143	27.14286	32.85714	45.35714	65	82.85714	91.07143	93.92857
	cf8	10.71429	16.07143	19.28571	26.78571	38.21429	48.92857	53.57143	55.35714
	cf9	17.85714	26.42857	31.78571	43.92857	63.21429	80.35714	88.21429	91.07143
	cf-combined	18.92857	27.85714	33.57143	46.42857	66.78571	85	93.57143	96.42857
Document level	WD-LBP	13.57143	21.07143	27.14286	37.14286	51.42857	66.78571	70	73.21429
	SIFT	16.07143	25	32.14286	43.92857	61.07143	79.28571	82.85714	85.71429

Table 6.4: consistency of features ranging from 25 to 280

	mean	std deviation	coefficient of variation	consistent features
gf1	61.9256	3.97323	6.41614	
gf2	68.8922	4.94295	7.1749	
gf3	85.0967	2.47127	2.90407	√
gf4	91.3211	5.41865	5.93362	√
cf1	76.6867	2.34397	3.05656	√
cf2	64.13	3.2226	5.02511	√
cf3	86.7733	2.04785	2.36	√
cf4	76.0189	3.05535	4.0192	√
cf5	96.1111	2.93303	3.05171	√
cf6	53.4467	3.61733	6.76811	
cf7	97.8911	2.14863	2.19492	√
cf8	61.8011	4.13255	6.68686	
cf9	94.5556	3.24502	3.43187	√
WD-LBP	85.1867	9.00296	10.5685	
SIFT	91.3132	4.90218	5.4643	√

Table 6.5: Performance at various levels

	Grapheme level				Character level				Document level			
	sensi–tivity	speci–ficity	precision	F measure	sensi–tivity	speci–ficity	precision	F measure	sensi–tivity	speci–ficity	precision	F measure
Naive bayes	0.8375	0.9995	0.8696	0.8388	0.8991	0.9997	0.9339	0.9028	0.7301	0.9992	0.7803	0.7386
k-NN	0.8714	0.9996	0.9	0.8734	0.9412	0.9998	0.9625	0.9463	0.7954	0.9993	0.8437	0.7986
SVM	0.8832	0.9996	0.9107	0.8860	0.9453	0.9998	0.9643	0.9526	0.8020	0.9993	0.8428	0.8082
Adaboost M2	0.8955	0.9997	0.9196	0.9018	0.9483	0.9999	0.9696	0.9533	0.8225	0.9994	0.8625	0.8271

Table 6.6: Decisive Features for Malayalam characters among all levels

	Grapheme level		Character level						Image processing based
Features	*gf3*	*gf4*	*cf1*	*cf3*	*cf4*	*cf5*	*cf7*	*cf9*	SIFT
Identification rate	82.14	84.29	72.85	82.5	70.71	92.14	93.92	91.07	85.714
Consistency	2.904069	5.933625	3.056556	2.359999	4.019199	3.0517	2.194923	3.43187	5.464301
Decisive Features	√					√	√	√	

6.5 Summary of the chapter

This chapter presented a mathematical model and discusses the experimental results obtained for the three schemes of writer identification. Certain important inferences were also derived. It was found that curvature and directional features can be used for writer identification with high accuracy. Stability and consistency of features were also analyzed.

Chapter 7

Application framework for Online Malayalam Character Recognition and Grantha script recognition

This chapter presents a system for online recognition of Malayalam and Grantha scripts. Grantha is an ancient language in India. This was once used to store and derive high level thoughts and knowledge linked with Sanskrit. Since there exists a close relationship between Malayalam and Grantha script, mapping of Grantha scripts to Malayalam is possible. Consequently the knowledge hidden in Grantha scriptures can be retrieved with the same system developed for online Malayalam character recognition. The system was designed of different phases like preprocessing, feature extraction, training and classification or recognition. Different feature extraction methods like context bitmap and time domain directional features are also put side by side for the purpose of comparison to obtain the efficient character recognition system. A full performance evaluation of both system using different classifiers in terms of recognition rate is discussed. The often misclassified characters are further explored in detail with the aid of confusion matrix and conclusions are drawn at. Two types of test beds were used in Grantha script recognition. One is manuscripts and the other is from Soundarya Lahari written by Sri Adi Shankara, a book written in Grantha script. The characters and words were recognized successfully in this test bed.

7.1 Introduction

Character recognition is the widest used application of pattern recognition [112]. While much work has been carried out to address this task for western languages, work on handwritten recognition for Indian languages is still lagging behind. The traditional mode of input/output such as a

keyboard is infeasible in case of Indic scripts. The development of pen interfaces is a key element which provides an efficient and natural way of human computer interaction.

To recognize characters written by humans, the first step is to convert them into digitized format recognizable by the computing machine. To obtain digitized version of the handwritten data,

i. The handwritten script can be scanned or

ii. Writing can be done with the aid of a special pen or with the help of a digitizer and liquid crystal display.

The two methods are commonly known as off-line and on-line handwriting respectively. Offline handwriting recognition focuses recognition of characters and words that had been recorded earlier which is prescribed to the system in the form of scanned image of the document. In contrast, online handwriting recognition focuses on tasks when recognition can be performed at the time of writing itself by tracking the coordinates of characters written by the writer as a function of time.

The recognition of handwritten characters in Malayalam is quite difficult due to the numerals, vowels, consonants, vowel modifiers and conjunct characters. The structure of the scripts and the variety of shapes and writing style of individuals at different times and among different individual poses challenges that are different from the other scripts and hence require customized techniques for feature representation and recognition.

The organization of the chapter is as follows. Section 7.2 describes related work. Section 7.3 portrays an overview of the Grantha Script. Section 7.4 points to the snaps of linkage between Grantha Script and Malayalam. Section 7.5 gives the overview of the system architecture in detail. Section 7.6 describes the comparative study in online Malayalam

character recognition. Section 7.7 depicts the framework recognizing Grantha script. The performance analysis is detailed in Section 7.8. The chapter is concluded in Section 7.9.

7.2 Related Work

Like many of the South Indian languages,Malayalm too has its roots in Indic script. Indic scripts have simple and complex characters. To analyze these different strategies have been used depending on its structural complexity. Characters have been viewed as compositions of strokes: This method required the knowledge of different strokes of a script in advance. Also it is found that strokes are based on the function of writing styles. This proves to be a key problem in this strategy. The analysis of training data was performed namely for the determination of unique strokes in studies [113], [114], [115], [116]. It was then estimated as 123 for Devanagari and 93 for Tamil [113]. These were approaches developed at IIT-M [113], [114], [115], [116] which are rule-based and script specific. Hence it is observed that significant manual intervention was required in the training phase.

Another approach was to view characters as a composition of C (An isolated consonant), C'C (A conjunct that combines two consonant sounds), V (An independent vowel) and M (modifier M to indicate nasalization of the vowel). The major advantage of this approach is that only a reduced effort is involved in data collection as only samples of identified graphemes are required. In the simplest form, this strategy does not address stroke order variations across symbols in the character. Symbol order variations co-articulation effects and the cases where opaque symbols which are created by CC and CV combinations are considered. Basic graphemes including core characters and ligatures, which come up to 141, are manually identified and modeled based on HMMs in Telugu script recognition [117].

These strategies were suited for complex characters but while analyzing simple characters the above would be far less efficient. Simple characters, such as isolated vowels and consonants in different Indic scripts, can be viewed as indivisible units. Standard character recognition techniques may be used without much knowledge of the structure of the script in the literature proves to be an advantage of this strategy. Joshi et al. [118] assume that Devanagari can be "linearized" like Tamil by constraining writers to unravel consonant clusters into sequences of vowel-muted consonant characters.

All studies show that the character recognition process can be divided into preprocessing, feature extraction and classification phases. Like other scripts, Indic scripts focus the similar preprocessing techniques. Dehooking, Smoothing, Resampling, Size normalization etc. are commonly performed. Different types of size normalization for Devanagari strokes are investigated in the Swethalakshmi's work [113]. As a way of designing an optimal scaling function that reduces classification error a Genetic programming approach has been explored in [119]. Nonlinear normalization which has been found to be effective for CJK (Chinese, Japanese and Korean) scripts does not appear useful for Indic scripts [113].

In the Feature extraction phase unique features of a script are analyzed. Latin scripts and Indic scripts have widely used low-level features such as normalized (x, y) coordinates. In addition to this first and second order derivatives of normalized (x, y) coordinates and curvature have shown promising results for Tamil and Telugu [120], [121]. Structural features such as cusps, bumps, loops and semi-loops have also been explored for Tamil [116], [122] and Devanagari [113] character recognition. Angle features are shown to be susceptible to noise leading to high intra-class variability whereas Fourier coefficients do not capture subtle differences between two Tamil characters. Wavelet features are shown to be the

most effective for Tamil as they retain both the intra-class similarity and inter-class differences. These features are compared with Neural Network classifier [123] for Tamil character recognition. In general, directional coding approaches popular for CJK (Chinese Japanese Korean) scripts are not effective for Indic scripts since the strokes do not have simple shapes. Rao and Ajitha [124] propose the use of (x, y) extrema, direction of pen motion (clockwise/anticlockwise) and relative displacement from the previous point of the same extrema category (x *or* y) for Telugu character recognition. Online features which model the input as a raster image rather than a trajectory may also be used to improve recognition accuracy when compared to using online features alone [125]. A combination of context bitmap and normalized (x, y) coordinates are used for online Malayalam character recognition [2].

In the classification phase a number of solutions in character recognition of Indic scripts are derived by using Template matching, rule-based, Neural Networks, Hidden Markov Models and subspacebased approaches. N.Joshi et al, [126] use template matching approach for writer dependent Tamil character recognition. Their approach is a two stage classification scheme of the combination of Euclidean distance and DTW distance has been used on Nearest Neighbor classifiers. By using a different set of features the same scheme has been applied for writer independent Telugu and Tamil character recognition [121]. Also, Rao and Ajitha [124] perform a coarse matching with the templates using the number of (x, y) extrema point using the Dynamic programming.

Rule-based approaches mainly exploit human knowledge about the problem since it does not have an explicit training phase. This has an advantage of requiring minimal training data whereas, it has advantage of suffering from being highly script-specific and labor intensive and it does not provide any alternate recognition choices. Strokes are first

classified using Support Vector Machines (SVM) and predefined rules are then applied for recognition of Devanagari character recognition [144]. In another work for the Devanagari script [127], each character represented by a set of combinations of stroke templates derived from the analysis of different writing style of Devanagari characters. But in Tamil character recognition an equivalent string of shape features is computed to recognize the unknown stroke using a flexible string matching algorithm [116].

In the work of Kunte and Samuel [128], Feed-forward Neural Networks with a single hidden layer are used for the recognition of handwritten Kannada characters using the feature vector of the coefficients of Wavelets from the preprocessed (x, y) coordinates. The work of Sundaresan and Keerthi [123] compares the performance of Time Delay Neural Network (TDNN) and a single hidden layer network for online Tamil characters. TDNN presents poor performance due to the presence of similar characters and high dimensionality of the input. A Multi-layer Perceptrons (MLP) based approach trained on eight-direction code histogram features is reported for Bangla character recognition [129]. Two-pass architecture recognition based on sequence of pen stroke using Back-propagation Neural Network is described by Jayababu et.al [130] for online Malayalam character recognition. The first pass does the initial classification and second recognition.

Only a limited work has been reported in Indic script based on Hidden Markov Model. The combination of HMM and Nearest Neighbor classifiers shows 86.5% of accuracy for Devanagari character recognition in Connell et al. [125]. A combination of time-domain and frequency domain features along with left-to-right HMM model shows a better accuracy in the character recognition of Telugu [117] and Tamil [120] scripts.

The subspace method for recognizing writer dependent Devanagari characters based on pre classified shirorekha and vowel modifiers using

heuristics approach is mentioned in the work of Joshi et al. [121]. Another work in Tamil character recognition [131] is based on the Eigen analysis on the feature vector of equi-spaced normalized (x, y) coordinates and has given better results by DTW based template matching [132]. Table 7.1 summarizes the online character recognition methods on multiple Indian languages.

7.3 Overview of Grantha Script

The Grantha script is evolved from ancient Brahmic script and it has parenthood of most of the Dravidian-south Indian-languages. In Sanskrit, 'Grantha' stands for 'manuscript'. As Grantha was used literally to 'transliterate' Sanskrit, the character set of Grantha closely resembles Sanskrit and is still used in traditional vedic schools. Grantha has a rich past during 5^{th} century AD. In 'Grantha', each of the letters represents a consonant with an inherent vowel 'a'. Other vowels are indicated using a diacritics or separate letters. Letters are grouped according to the way they are pronounced. The major derivations of 'Grantha' are Archaic Grantha, Transitional Grantha, and Modern Grantha. Archaic Grantha is the primitive form and were used by Pallavas. For the inscriptions, an ornate type of 'Grantha' was used. Major transcriptions are Tiruchirappalli Rock cut cave, Kailasanath and Mamallapuram inscriptions. 8^{th} century derivation of 'Grantha' is 'Transitional Grantha' and Malayalam is closely connected to it. The Thulu-Malayalam script of the Grantha is in two forms: the Brahmanic (square) and Jain (round). Pandiyan Nedunchezhiyan inscriptions are the examples of it. Modern Grantha has got its development from the time of the inscriptions of Thanjavur cholas to the Vijayanagara rulers.

Table 7.1: Online character recognition methods on multiple Indian language

System	**Number of Classes**	**Features**	**Classification**	**Accuracy**
Devanagari				
Swethalakshmi 2007	123 strokes	X-Y coordinates, Fourier descriptors and structural features	SVM and rule-based stroke re-grouping	89.88%
Connell et al. 2000	40 simple characters	Online and oine	Combination of HMM and Nearest Neighbor classifiers	86.5%
Tamil				
Ramakrishnan et.al 2007	156 characters	Structural features	Hierarchical Classification	96%
N. Joshi, G. et.al. 2004	156 characters	X-Y coordinates	Nearest Neighbor Classifier using Dynamic Time Warping (DTW) distance	91.20%
Toselli et al. 2009	156 characters	Time-domain and frequency-domain	HMM	90.72%
Telugu				
Prashanth et al. 2007	141 core characters and ligatures	X-Y coordinates, normalized first & second derivatives and curvature	Nearest Neighbor Classifier using Euclidean and DTW distance	89.77%
Babu et al. 2007	141 core characters and ligatures	Time-domain and frequency-domain	HMM	91.6%

Online character recognition methods on multiple Indian language

System	**Number of Classes**	**Features**	**Classification**	**Accuracy**
Bangla				
Bhattacharya et al. 2007	50 simple characters	8-direction code histogram	MLP	83.61%
Malayalam				
G. Jayababu , et al. 2004	64 characters	8-direction	Back-propagation Neural Network	85.3%
M. Sreeraj, et al. 2009	64 characters	context bitmap and normalized (x, y) coordinates	Self-organizing maps (SOM)	88.75 %
M. Sreeraj, et al. 2010	64 characters	Time Domain and Directional Features	k-Nearest Neighbor Classifier	98.125%

There are 14 vowels. Of these 7 are the basic symbols. Long vowels and diphthongs are derived from these. Also there exists 13 vowel modifiers, and there are no full vocalic short *l* and full vocalic long *l* modifier. Grantha admits 34 basic consonant characters. As with all Brahmi derived scripts, the consonant admits the implicit vowel 'schwa'. Pure consonant value is obtained by use of the virma. Grantha has two diacritic markers: the anuswara and the visarga. The anuswra is a latter addition and in Archaic as well as Transition Grantha the letter ma is used to represent the nasal value. A special feature of Grantha is the use of subsidiary symbols for consonants. These are three in number: the use of a subsidiary 'ya' and two allographs for ra depending on whether ra precedes the consonant or follows it [150]. The Fig 7.1 gives the symbols used in Grantha script and Fig 7.2 depicts the taxonomy of Grantha script [151].

The consonant 𑌪 is represented in two ways. When following a consonant it is written as 𑍍𑌯 under the consonant; but when it precedes a consonant it takes the 𑌰𑍍 form written after the consonant or conjunct.

Complex consonantal clusters in Grantha script use the *Samyukthaksharas* (Conjunct characters) widely. Combined with vowel signs, these *Samyukthaksharas* are considered as a single unit and placed with the Vowel signs. The *Samyukthaksharas* of Grantha is formed in the following three ways [152].

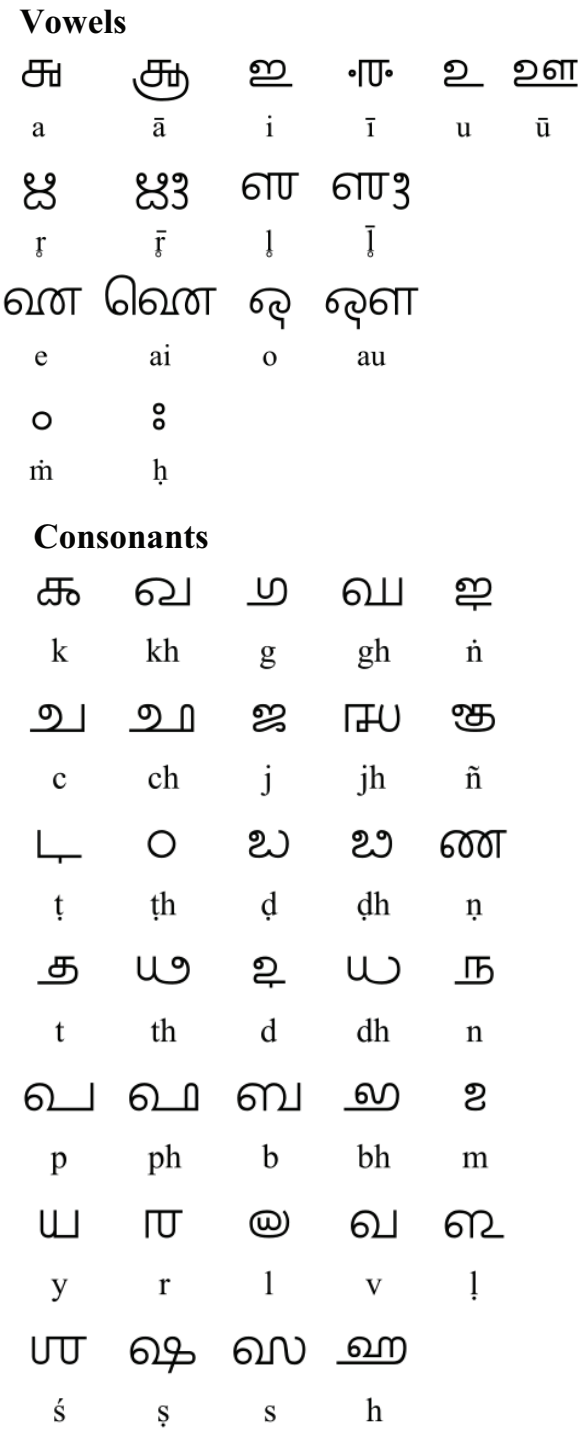


Figure 7.1: Grantha characters

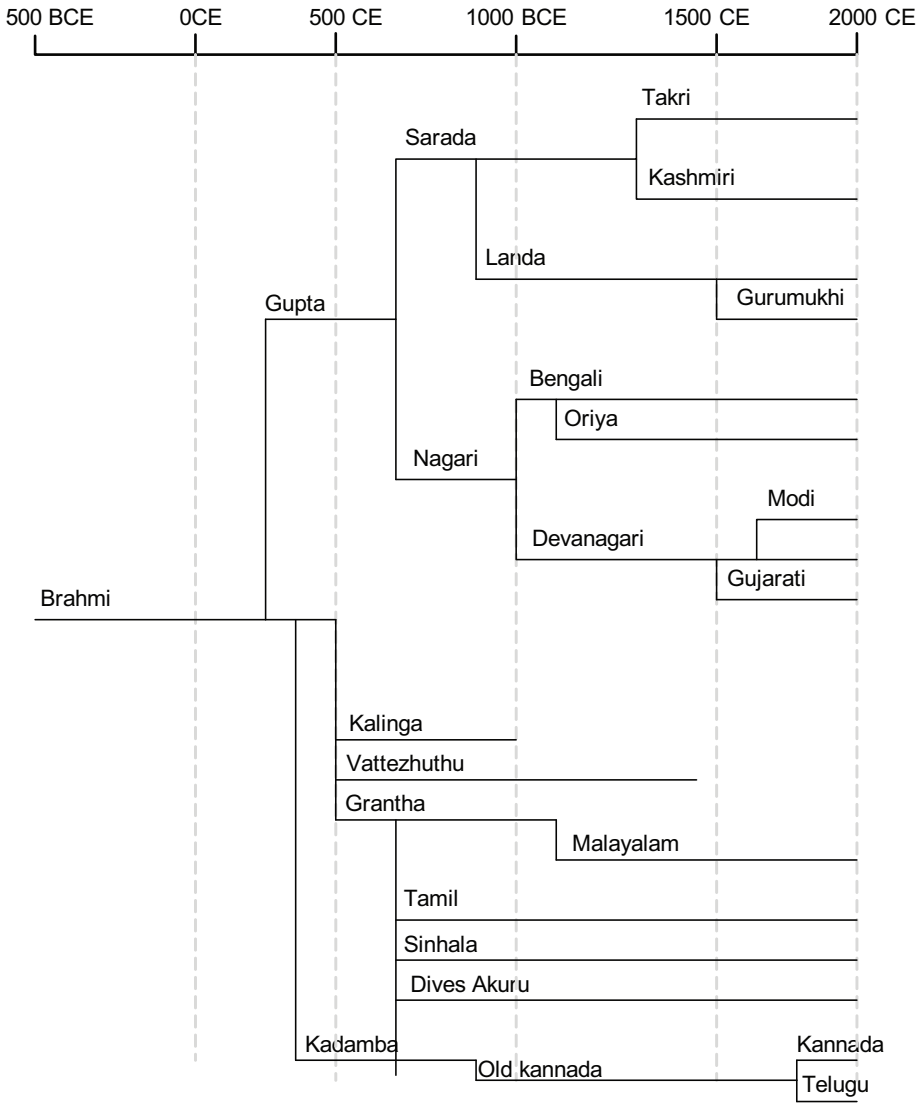
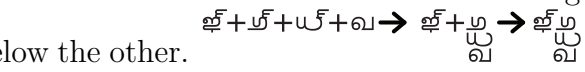


Figure 7.2: Taxonomy of Grantha script

7.3.1 Stacking

Stacking the consonants on each other to form *Samyukthaksharas* is a common method, but these consonants must be clustered first, and as required it should be stacked with the remaining third consonant. In Grantha the stacking is limited to three and also a single written "akshara" is considered by clustering conjunct formation of the combining conjuncts. Here the first consonant will be in full width and other two following consonants as miniatures one below the other. 

7.3.2 Combining

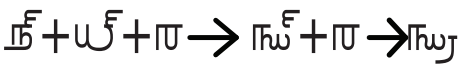
Fusion of consonants to make a combined single unit and forming a *Samyukthaksharas* is another method. Two or more conjunctive consonants makes a ligature with an individual shape.

7.3.3 Special signs

Adding a special signs of a consonant, to another consonant is lieu of fusing independent consonants to form a *Samyukthaksharas* is the third method. In this case a special signs such as:  and  may be used to represent a consonant to be added with another consonant.

-r- Conjunct

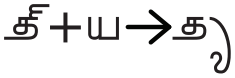
There are 2 types in this method; those are post and pre consonantal forms- In post consonantal form, unlike other consonants, it morphs itself into a special character when -r- occurs as the final part of the consonantal cluster. In complex form when -r- occurs as the final consonant in a triple cluster the miniature version of the *Samyukthaksharas* sign is placed by stacking conjuncts.



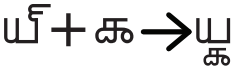
In pre-consonantal conjuct form it will be placed next to the consonant when -r- appears in the first consonant of the character. It also has a complex form, in case of the final consonants barring -r- is clustered first where the pre-consonantal -r- sign is followed when triple clusters appear with pre-consonantal -r-.

-y- Conjunct

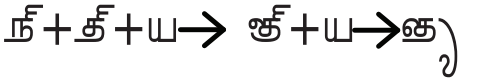
When appearing in the post consonantal position of a cluster -y- has also a certain form like -r- conjuncts.



But the cluster character is normal like stacking as /ya/ appears as the first consonant.



By considering special cases using the stacking or combining rules the first two consonants must be first clustered and then /ya/ sign should be added for triple cluster *Samyukthaksharas* with -y- as the final element.



-ry- Conjunct

The *Samyukthaksharas* -rya is formed by giving priority to the pre-consonantal form of -r- rather the post consonantal form of -y- as discussed earlier. For complex conjuncts special signs are placed next to each other.

7.4 Grantha Script and Malayalam - Snaps of Linkage

The foundation of Malayalam script owes itself to Grantha script. They have much similarity with Grantha scripts. When Grantha scripts were used to write Sanskrit letters, it was called Kolezhuthu (rod script).Complex Combining Conjuncts with 4 (or more) consonantal clusters are present in Grantha but in old Malayalam script it is only up to 2 consonantal clusters. The special vowelless forms of these consonants ൟ & ൮, ൣ & ൯ is also seen rarely in Grantha. The other consonants do have forms like this, which can be found in manuscripts. However they are not found in any printings. Number of vowels are decreased by the absence of characters corresponding to ൠ(ൃ),ൡ(ൄ),൰(ൗ) . Complex stacking conjuncts are present in Grantha and it is absent in the new script of Malayalam.

7.5 Generic System Architecture

This system is developed by using the elegant feature obtained from the previous chapter ie, writing direction and curvature. It is observed that this feature is more convenient and stable for online Malayalam character recognition while this has been compared with the systems having context bitmap[2] and time domain geometrical features with kohonen SOM. The Fig.7.3 describes the generic system architecture for online character recognition for Malayalam.

7.5.1 Pen device and Data sets

In the experiment the input strokes are read using pen device Wacom Graphire 4 CTE-640 and are stored in UNIPEN format [133] [134]. UNIPEN format of a character taken as a sample is shown in the Appendix

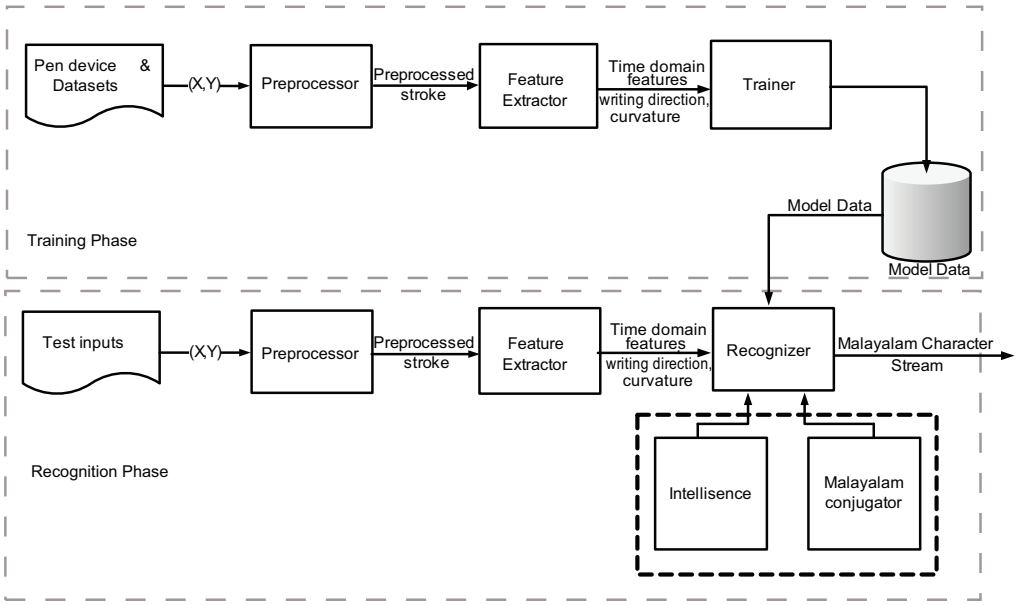


Figure 7.3: System Architecture

G. To ensure maximum accuracy the database of 80 writers have been collected.

7.5.2 Pre-processing

The raw data collected is preprocessed to reduce spurious noises like dots and stray strokes. The strokes are smoothened using a large filter so that unwanted cusps and intersection are removed. Malayalam scripts essentially require this phase as it is curvaceous in nature. To remove jitter from the handwritten text, we replace every point $(x(t), y(t))$ in the trajectory by the mean value of its neighbors:

$$\mathrm{x}'(t) = \frac{x(\mathrm{t}\ -\ \mathrm{N}) + ...\ +\ \mathrm{x}(\mathrm{t}\ -\ 1)\ + \alpha x(t) + \mathrm{x}(\mathrm{t}\ +\ 1) + ...\ +\ \mathrm{x}(\mathrm{t}\ + \mathrm{N})}{2\mathrm{N} + \alpha} \tag{7.1}$$

And

$$\mathrm{y}'(t) = \frac{y(\mathrm{t}\ -\ \mathrm{N}) + ...\ +\ \mathrm{y}(\mathrm{t}\ -\ 1)\ + \alpha y(t) + \mathrm{y}(\mathrm{t}\ +\ 1) + ...\ +\ \mathrm{y}(\mathrm{t}\ + \mathrm{N})}{2\mathrm{N} + \alpha} \tag{7.2}$$

The parameter α is based on the angle subtended by the preceding and succeeding curve segment of $(x(t), y(t))$ and is empirically optimized. This helps to avoid the smoothing of sharp edges, which provide important information when there is a sudden change in direction [135]. To normalize the data scaling has been adopted as it ensures each handwritten characters have no canonical representation thereby ensures that size makes no difference for recognition.

Dot detection: Any stroke, if it has to be considered as a dot should be below the normalized dot size threshold. The threshold value is 0.01 and it is expressed in real length terms (inches) and converted internally to

points using the knowledge of the device's spatial resolution. If the width and height is both less than this threshold then it treated as a dot.

Dehooking: Dehooking is done to eliminate stray strokes that appear due to inaccuracies in pen down position or rapid erratic motion in placing the stylus on the tablet. Hooks are positioned at the ends of a stroke accompanied by a sharp turning point in the stroke. It can be detected by checking the changes of the turning angle as well as the location. Turning angle is formed by consecutive line $\overline{P_{i-1}P_i}$ *and* $\overline{P_iP_{i+1}}$ (Fig 7.4). mathematically, the following conditions are used to detect hooks:

- $\phi_i > \theta$ Where $\theta$is a given threshold.
- $\sum_{k=i}^{n-1} arcLength(P_{k+1}, P_k) < \alpha L$, where α is a real number between 0 and 1and L is the stroke's length. In this system $\theta = 90^o$ and $\alpha = 0.13$.

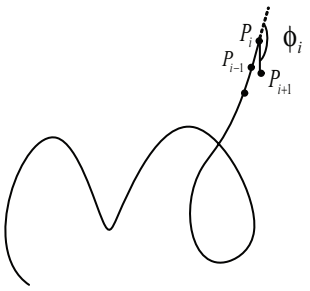


Figure 7.4: Dehooking on character 'n'.

7.5.3 Feature Extraction

This is the module where the features of handwritten characters are analyzed for training and recognition.

Time domain geometric features

The Fig.7.5 represents a sample stroke together with values of selected features (time domain [139], [140], writing direction and curvature [141]). A continuous stroke of a character can be described by means of a set of discrete multidimensional vectors $C_i,\ i = 1,\ n$ (Fig.7.5). Any vector C_i has f coordinates, where f depends on number of parameters which should be analyzed. Hence, a character stroke description consists of nXf real numbers, stored in a text file. Let C be a given character, consisting of $C_i, i = 1 \ldots n$ points. In proposed approach, it is assumed that any point can be described with following features.

- **Normalized x-y coordinates**: The x and y coordinates from the normalized sample constitute the first 2 features.
- **Pen-up/pen-down**: Pen-up/pen-down feature is dependent on the position sensing device. The pen-down gives the information about the sequence of coordinates when the pen touches the pad surface. The pen-up gives the information about the sequence of coordinates when the pen not touching the pad surface. Calculate the time sequence of each pen-down and pen-up between every stroke.
- **Velocity**: by using the following formula, the velocity has been measured by taking the average velocity between several points of each stroke.

  $$vx_i = \frac{x_{i+1} - x_i}{t_{i+1} - t_i}, vy_i = \frac{y_{i+1} - y_i}{t_{i+1} - t_i}$$

  for i=1.....n-1
- **Writing direction**: The local writing direction at a point $(x(n), y(n))$ is described using the cosine and sine [140].

  $$sin\theta(n) = \frac{Y_n - Y_{n-1}}{\sqrt{(X_n - X_{n-1})^2 + (Y_n - Y_{n-1})^2}} \tag{7.3}$$

$$cos\theta(n) = \frac{X_n - X_{n-1}}{\sqrt{(X_n - X_{n-1})^2 + (Y_n - Y_{n-1})^2}} \tag{7.4}$$

- **Curvature**: The curvature at a Point $(x(n), y(n))$ is represented by $cos\phi(n), sin\phi(n)$. It can be computed using the following formulae.

$$Sin\phi(n) = Cos\theta(n-1) \times Sin\theta(n+1) - Sin\theta(n-1) \times Cos\theta(n+1) \tag{7.5}$$

$$Cos\phi(n) = Cos\theta(n-1) \times Cos\theta(n+1) + Sin\theta(n-1) \times Sin\theta(n+1) \tag{7.6}$$

Taking into account above considerations character C from the Fig. 7.5 can be described by the set of the multidimensional vectors:

$$C = \{c_1, c_2, ...c_n\} \tag{7.7}$$

Where: $c_i = \{x, y, \Delta t, v_x, v_y, \sin\theta, \cos\theta, \sin\phi, \cos\phi\}$

for $i = 1 \ldots n$ The elements of this vector will be the features for training and recognition modules.

7.5.4 Character Training and Recognition

The feature extraction module resulted in a combination of different features. In the training module these feature vectors and class labels of the training samples are trained and stored as model data. In the recognition module, the same features as before are computed for the test samples. Recognition function predicts the class label of the test sample by the basis of similarity measurement with the model data. The learning technique k-NN (K-Nearest Neighbor Classifier) is used in this system for training and recognition.

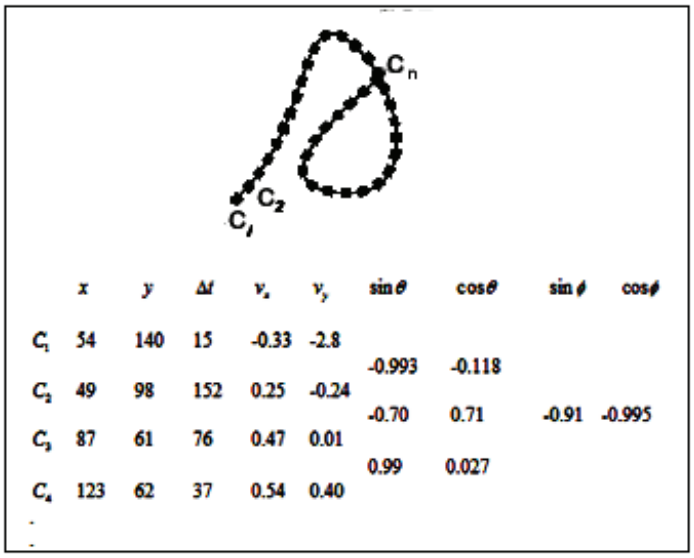


Figure 7.5: Sample stroke

7.5.5 Implementation

The system was implemented in JAVA using JNI,compatible for a 32-bit AMD Athlon 2.0 with 512MB RAM. Steps in training and recognition phases are described in Algorithm 8 and 9.

To incorporate intellisense feature a dictionary with 2000 Malayalam words was created which will be loaded while recognition. A popup window displaying Malayalam words from the dictionary using a binary searching algorithm was implemented.

7.6 Choice of features and classifiers

Comparative experiments have been taken place with different observations of features and classifiers. The system of the combination of Time domain geometric features and the k-NN classifier is compared with the system

Algorithm 8 Training Phase

1. Read character data using a capturing device
2. Load a property file containing parameters DotThreshold, DehookThreshold, LoopThreshold, Preserve AspectRatio, ResampleDimension, PrototypePerClass, PrototypeDistance.
3. Preprocess data using the parameters in property file, mentioned in step 2.
4. Calculate feature vectors of each re-sampled point of every character
5. For training, map each sample stroke of every character with corresponding class label. Result is a list file.
6. Select a prototype from each class and the resultant file is a model file, which is the input for recognition.

Algorithm 9 Recognition Phase

1. Load the property, list and model files in memory.
2. Calculate the feature vector of the sample
3. Find out the nearest class label of the sample using chi-square distance measurement.
4. Retrieve the Unicode of the Malayalam character corresponding to the class label and display it.
5. Check the Unicode obtained against a rule base to know whether it is a part of a conjunct letter.
6. If the output of step 5 is "YES' replace the two or more Unicode characters with the corresponding conjunct letter as defined in the rule base.
7. If the answer to step 5 is "NO', display the character corresponding to the retrieved Unicode.

of the combination of Time domain geometric features and the kohonen (SOM) network which is found competitive when compared to the system of context bitmap and kohonen (SOM)[136]. It is found that k-NN provides better results and hence selected for implementation.

7.7 Grantha script recognition

It has been observed previously that the designed framework is most efficient in online character recognition of Malayalam. Because of the merit of the framework, with an additional module of Grantha conjugator, here it describes an extension of the same framework. What differentiates in the online recognition of Grantha scripts is that unlike in Malayalam characters, up to four levels of stroke combinations of characters can be recognized. Writing over the characters in a manuscript or in any other medium being identified in this process, and the hidden knowledge of this extinct language can be retrieved and transferred to Malayalam. Out of the five modules of the framework, the first module preprocesses the datasets, to necessitate the way for feature extraction. The second module is the feature extraction module. The features extracted here are time domain features based on writing direction and curvature, which is discussed in the section 7.5.3. The next two modules are part of the classification process namely trainer and recognizer. The knowledge feature vector of the model data from the trainer is fed as one of the inputs to the recognizer in the testing phase. The recognizer is aided with the Grantha conjugator, which could extract the rules for conjunct characters. The fifth module is the converter, where the convertion of Grantha script to old and new Malayalam characters is done. The converter is supported by intellisence feature, which is incorporated to avoid the problem corresponding to the absence of certain characters in new script of Malayalam by providing the equivalent word by searching from a dictionary. Also Malayalam conjugator aided the converter to form

rules for conjunct characters in order for Grantha script to be converted into Malayalam.

The classifier used here is k-NN. Dynamic Time Warping (DTW) distance is used as the distance metric in the k-Nearest Neighbor classifier. Dynamic Time Warping is a similarity measure that is used to compare patterns, in which other similarity measures are practically unusable. If there are two sequences of length n and m to be aligned using DTW, first a nXm matrix is constructed where each element corresponds to the Euclidean distance between two corresponding points in the sequence. A warping path W is a contiguous set of matrix elements that denotes a mapping between the two sequences. The W is subject to several constraints like boundary conditions, continuity, monotonicity and windowing. A point-to-point correspondence between the sequences which satisfies constraints as well as of minimum cost is identified by the following formula.

$$DTW\,(Q,C) = \min\left\{\sqrt{\sum_{k=1}^{K} w_k}\;/K \right. \tag{7.8}$$

Q, C are sequences of length n and m respectively. W_k element is in the warping path matrix. The DTW algorithm finds the point-to-point correspondence between the curves which satisfies the above constraints and yields the minimum sum of the costs associated with the matching of the data points. There are exponentially many warping paths that satisfy the above conditions. The warping path can be found efficiently using dynamic programming to evaluate a recurrence relation which defines the cumulative $distance(i, j)$ as the $distanced(i, j)$ found in the current cell and the minimum of the cumulative distances of the adjacent elements. [131].

The online character recognition for Grantha script is achieved through

two steps i) Training and ii) Recognition. Training of the samples is done setting the prototype selection as hierarchical clustering. Recognition function predicts the class label of the input sample by finding the distance of the sample to the classes according to the K-Nearest Neighbor rule. DTW distance is used as the distance metric in the k-Nearest Neighbor classifier. The top N nearest classes along with confidence measures are returned as the identified letters and the top most one will be chosen as the recognized character.

The recognized Grantha scripts are converted to the equivalent in Malayalam to form a meaningful word considering even the different variants in Malayalam. Algorithm 10 explains the conversion of Grantha word to Malayalam.

7.8 Performance Analysis

The success of handwritten recognition system is vitally dependent on its acceptance by potential users. The system was tested according to two schemes such as writer dependent and writer independent. Writer dependent testing was named Scheme1 while writer independent testing was named as Scheme 2. Writer-dependent system provides a higher level of recognition accuracy than writer independent system. The amount of training data that must be supplied by the user before the system can be used may impede its acceptance in writer dependent system. On the other hand, a writer independent system must be able to recognize a wide variety of writing styles in order to satisfy an individual user.

For training phase 20 samples each of 80 writers for 64 basic characters were collected and the system was trained. Therefore a total of 204800 samples were collected for training. The systems were further tested using scheme 1 and scheme 2 as detailed above. For test of scheme 1 writings of

Algorithm 10 Conversion of Grantha word to Malayalam

Input: Grantha Word W_g
Output: Malayalam Word W_{mo} and W_{mn} in old script and new script respectively

for each character $w_g(i)$ *in* W_g
  if ($w_g(i)$==ൻ||ൽ)
  then
    temp_store = $w_g(i-1)$;
    $w_g(i-1) = w_g(i)$;
    $w_g(i)$=temp_store;
  end
end
for each character $w_g(i)$ in W_g
  Set char_type;
  if char_type= = vowel or char_type==consonant
    Set $w_{mo}(i)$ and $w_{mn}(i)$ directly
  else if char_type == complex stacking form of conjunct letters
    Find R from conjugator// example of R=ക്ഷ→ക+ഷ്→ ക+ഷ്+ഷ)
    Set $w_{mo}(i)$ and $w_{mn}(i)$
  else if char_type==combined form of conjunct letters
    Find R' from conjugator//example of R'=ങ്ക→ങ+ക്→ ങ+ക്+ക്)
    Set $w_{mo}(i)$ and $w_{mn}(i)$
  end
$W_{mo} = Concatenate(w_{mo}(i))$;
$W_{mn} = Concatenate(w_{mn}(i))$;
End

The complexity of the above algorithm is $O(n)$.

40 people whose samples were also used for training were put to test. In the test of scheme2 additional writings of 40 new people were also put to test.

After conducting the test schemes 1 and 2, it was notified that some characters had frequently erroneous nature due to their similarity. Fig. 7.6 shows the misclassified characters. For a better quality of result more number of samples of frequently erroneous characters were also included in the training set.

Figure 7.6: Misclassified characters

A graphical analysis of the performance of the systems used in the comparative study under the category of writer dependent and writer independent character recognition schemes was done.

The two features described earlier as context bitmap and time domain and directional features were extracted and was classified using Kohonen Networks. The experiments resorted to find out the influence of number of nodes in the Kohonen network, on the recognition rate and graph were plotted between number of nodes and recognition rate in Fig. 7.7. It is obvious that the recognition rates increase rapidly with increase in number of nodes and reach a maximum at 500, and then remains constant.

Hence the number of nodes is retained as 500 for the purpose of character recognition. The second experiment was attempted to investigate the effect of the number of iterations on the recognition rate. The dimension of the feature vector was fixed and the number of nodes was fixed to 500. In Fig.7.8 the recognition rates increases rapidly, with increase in number of iterations and reaches a maximum of 100, and thereafter remains approximately constant. Hence the number of iterations was decided as 100. The overall performance feature vector of a combination of context bitmap and normalized (x, y) coordinates was 88.75 %. In the system of time domain and directional features were exploited and the overall accuracy was found to be 92.25%. This clearly shows that the time domain and directional features is better suitable for on line Malayalam character recognition.

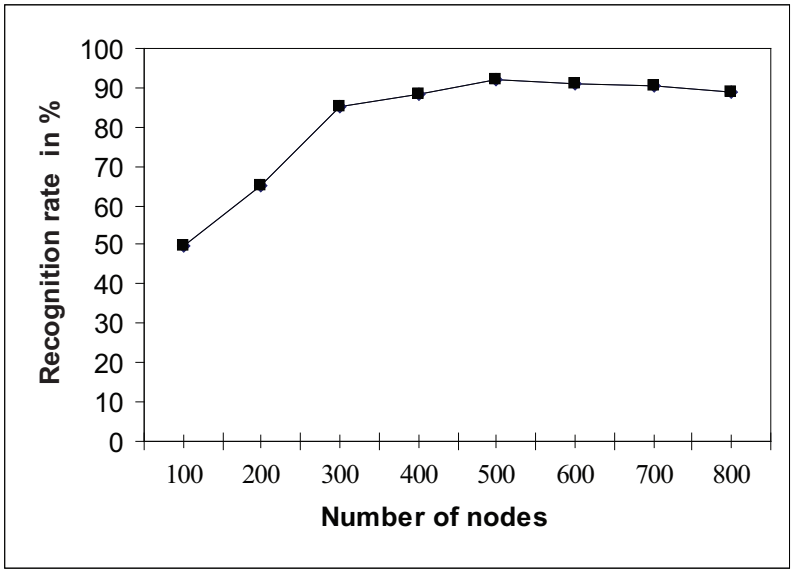


Figure 7.7: Recognition rates and Number of nodes

After conducting the test schemes 1 and 2 on the test bed of the system

of the combination of k-NN classifier and Time domain geometric features where the optimum value obtained for k was 9. This was found that the best performance for scheme1 and scheme2 were 99.68% and 98.13% respectively. The overall performance of different systems is summarized in Table. 7.2

The graph (Fig.7.9) compares the performance of the classifiers before and after the inclusion of similar characters in the training set. It is evident that the presence of similar characters in the dataset is highly influencing the recognition rate. Hence the confusion matrix for misclassified similar characters in each type of classifier is also arrived at in Fig.7.10

The experiment proceeded to recognize the scripts of Grantha yielded good result. From the chosen manuscripts the total of characters used in the test bed is 26712. Each manuscript is having 306-504 characters as they

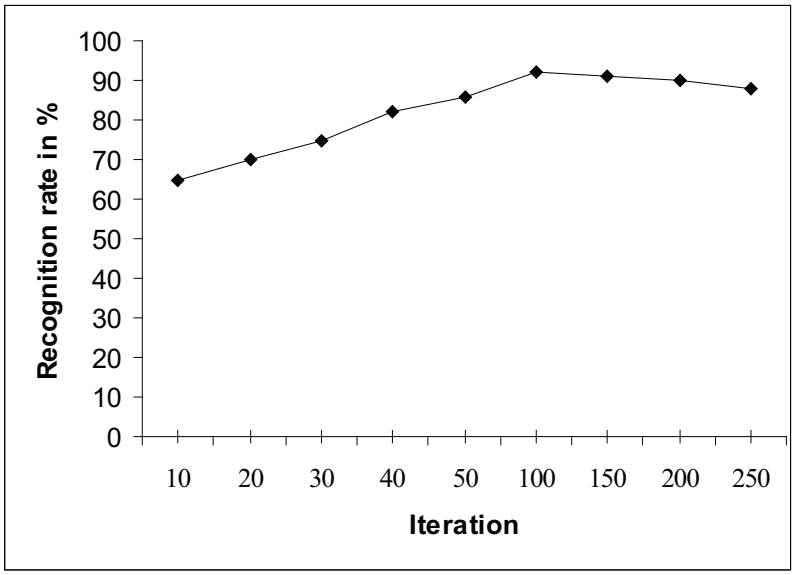


Figure 7.8: Recognition rates and Number of Iterations

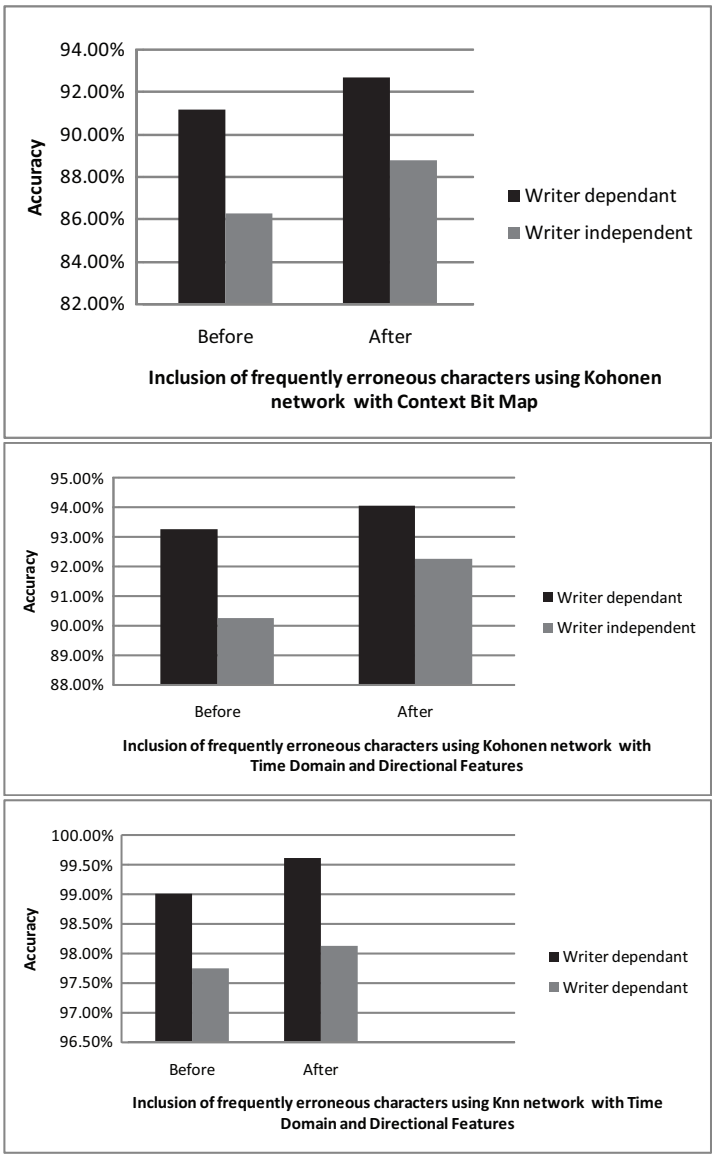


Figure 7.9: Performance of the classifiers before and after the inclusion of similar characters in the training set

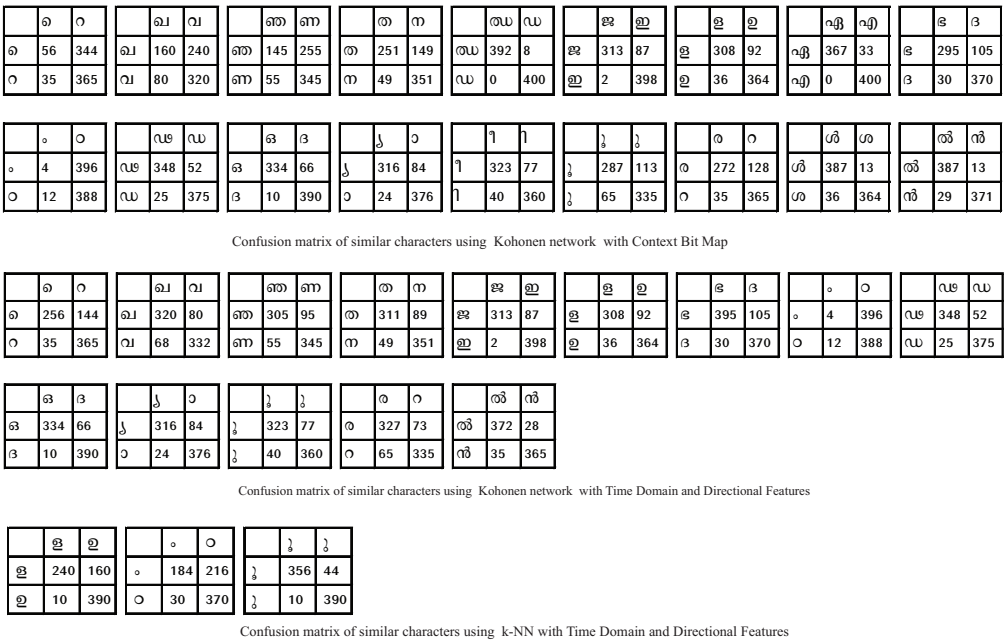


Figure 7.10: Confusion matrix for misclassified similar characters

Table 7.2: overall performance of different systems

Sl. No.	Feature Vector	Classifier	A/B *	Writer Dependent	Writer Independent
1	Context BitMap	Kohonen	A	91.17%	86.25%
	Context BitMap	Kohonen	B	92.65%	88.75%
2	Time Domain and Directional Features	Kohonen	A	93.25%	90.27%
	Time Domain and Directional Features	Kohonen	B	94.05%	92.25%
3	Time Domain and Directional Features	k-NN	A	99.02%	97.75%
	Time Domain and Directional Features	k-NN	B	99.63%	98.13%

are having 9-12 lines of 34-42 characters. Different distance measurements were applied in the classifier in order to distinguish the most accurate one. Fig.7.11.shows the recognition rates for different distance measurements of different groups of letters. Experiments were conducted on the frequently misclassified characters to find their similarity and their confusion matrix is shown in Fig 7.12.

The correct word limiter space in manuscript was not able to be recognized and it is resolved by searching the possible word from the dictionary while making the conversion between Grantha and Malayalam. In the word recognition process, 3180 words from the manuscript and 86390 words from a book pages were put into test. The chosen book was Sri Adi Shankara's 'Soundarya Lahari', printed in Grantha script. Each page of it contained 32-35 lines consisting of a total of 160-190 words. The test was conducted only to recognize 455 lines due to the availability of limited

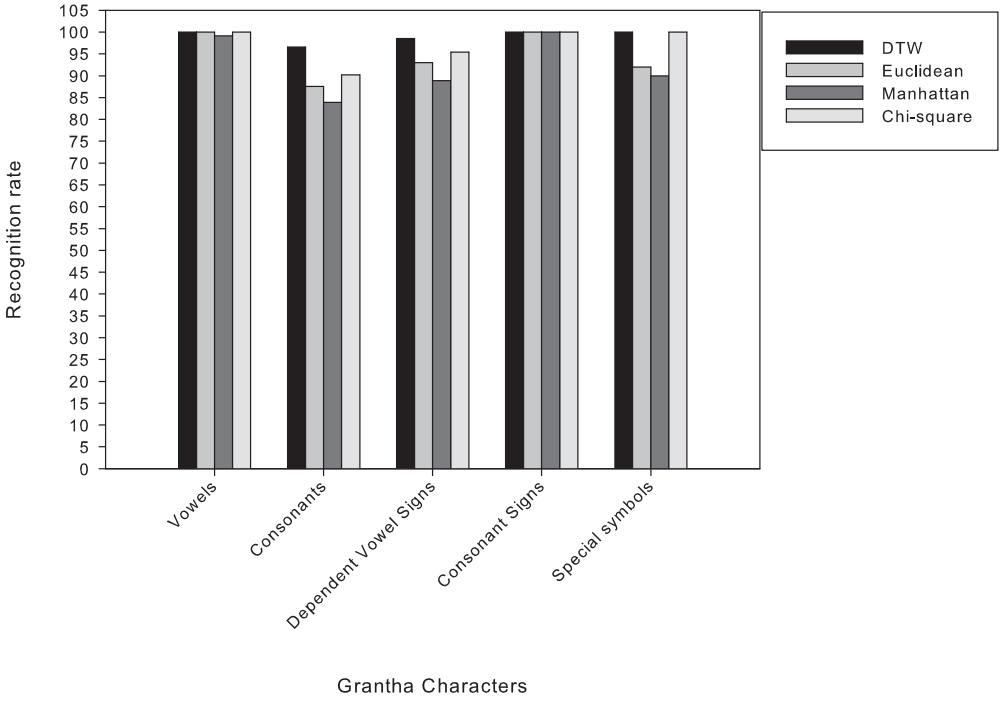


Figure 7.11: Recognition rates for different distance measurements

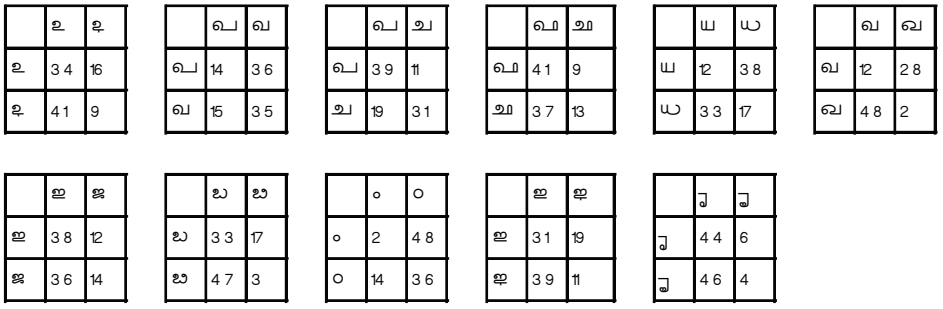


Figure 7.12: Confusion Matrix of Frequently misclassified characters

number of pages. Table 7.3 summarizes the result of recognition rate of words.

Table 7.3: Recognition rate of Grantha words and that of Malayalam

	Grantha	old Malayalam	new Malayalam
manuscript	92.11%	90.82%	89.56%
book	96.16%	95.22%	92.32%

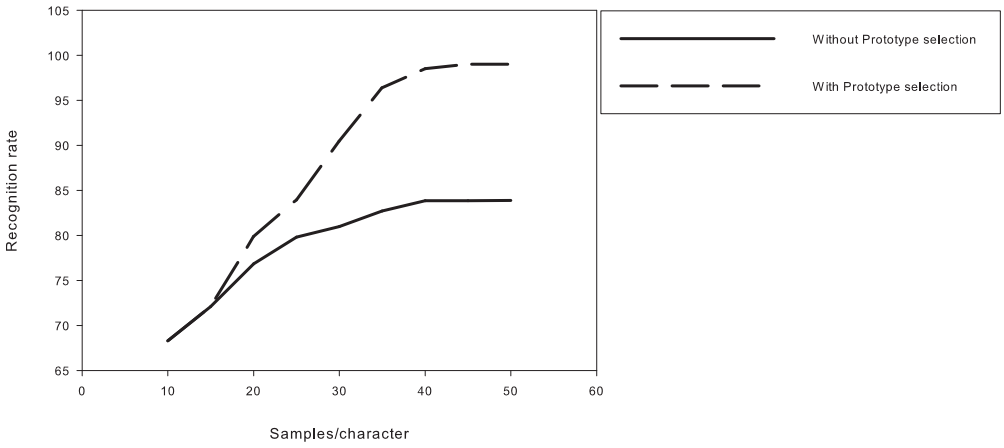


Figure 7.13: Recognition rate with and without Prototype Selection

Training of the samples was done with or without prototype selection, and the same is compared in terms of recognition rate in Fig. 7.13. It is evident that the samples trained with prototype selection paved way for better recognition rates. From the Table 7.3 it is clear that some variation in the recognition rate of the words in the different form of Malayalam scripts happened by the error occurred in the rule base structure of the

converter. So the test was also done across the characters on the false words generated by the converter to identify where the error occurred. And it shows that 93.5% misclassification occurred in the conversion of the stacked character in the Grantha script.

During the above experiments the misclassified words were noticed and test was conducted on these misclassified words to find the module where the error occured i.e., in the conversion phase or in character recognition phase. To minimize the error rate of misclassified words in the recognition phase, more number of samples of strokes of common character of the misclassified words were included in the training set. The yielded result was a reduced error rate from 4.16 to 1.86. Fig 7.14 shows the error rate reduction using two prototype selections. It is observed that the conversion phase error is mainly due to the absence of some characters in the new generation Malayalam for which more equivalent words are included in the dictionary.

The recognition rate for the different symbols in Grantha Script was tried with classifiers like k-NN and SVM. Specific experiments were done with and without DTW algorithm using k-NN classifier. A bar graph is plotted in Fig 7.15 with different classifiers and recognition rates. In Fig7.16 the recognition rate for different symbols is analyzed with different kernel functions like sigmoid, polynomial and RBF for SVM classifier.

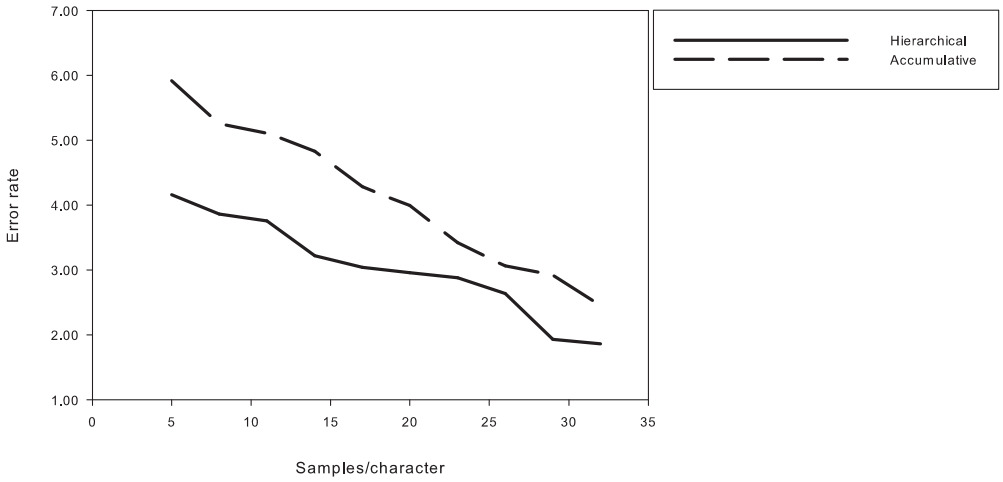


Figure 7.14: Comparison of error rate using Hierarchical and accumulative prototype selection methods

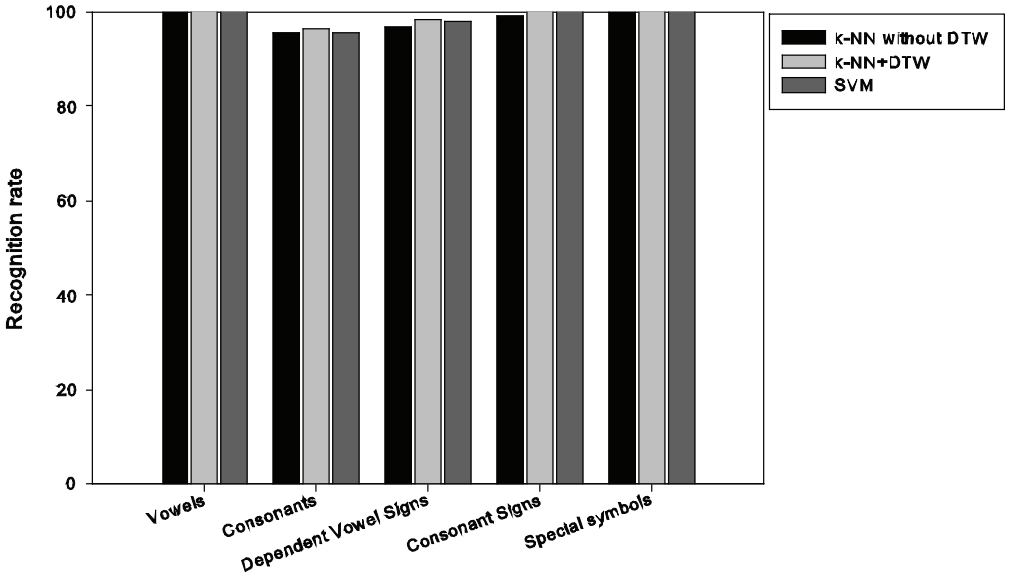


Figure 7.15: Category-wise comparison of recognition rates using different classifiers

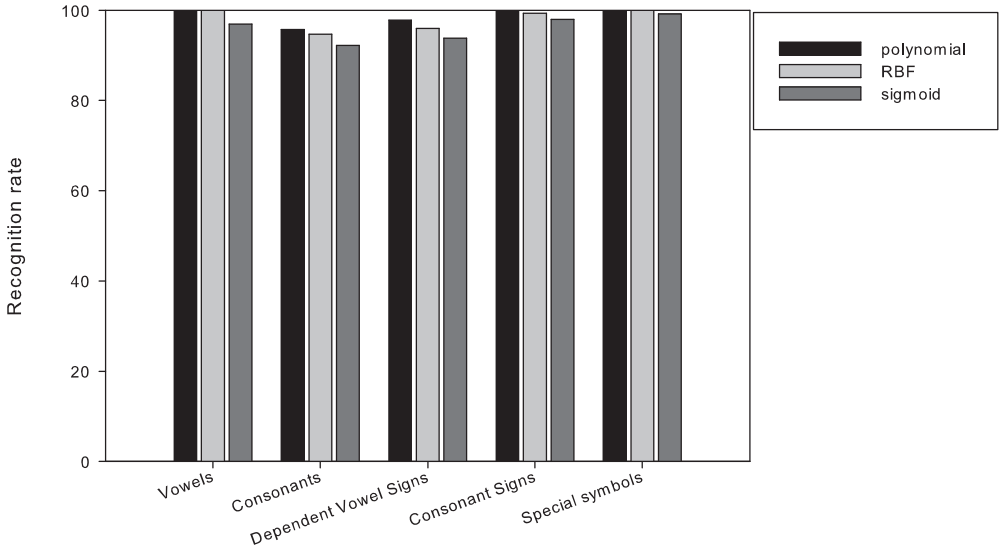


Figure 7.16: Recognition rate with respect to different kernel functions for SVM classifier

7.9 Summary of the chapter

To conclude, this chapter presents a framework for online character recognition system for Malayalam characters with fairly considerable recognition rate and further developed to recognize Grantha scripts. Here it reaffirms that the directional and curvature feature is most suitable for Malayalam characters. The systems in use are the combination of time domain geometric feature with k-NN, time domain geometric feature with kohonen SOM and context bitmap with kohonen SOM where the high performance comes from former to latter. Other major concern is to obtain the confusion matrix at the misclassification caused by the similar characters. The same framework having directional and curvature and k-NN classifier is used for the Grantha script. The framework was tested with manuscripts as well as a book. Different distance measures for evaluating the similarity were tried, of which DTW method showed considerable recognition rates. It has been observed that there are frequently misclassified Grantha characters. They have been studied in detail with the aid of confusion matrix and conclusion is drawn that the error is mainly occurring in the conversion phase. To reduce the error rate, the training of sample strokes is done using two prototype selections. Also the recognition rate is evaluated against two classifiers, namely, k-NN and SVM classifier. Further it is understood that the system of directional and curvature feature and k-NN classifier is the most suitable one for Malayalam and Grantha scripts.

Chapter 8

Conclusions and Future Directions

8.1 Conclusions

This thesis primarily addressed the problem of automatic writer identification of handwritten Malayalam documents. Malayalam through its inherent scripting technique, presented many challenges, which resulted in a detailed analysis and experimentation with different types of feature vectors. The features which were experimented with were at grapheme, character and document levels. The features identified at these three levels were put analysised using different classifiers. Experiments were conducted to obtain the performance of individual and combined features to come to an important conclusion that the combined features outperform the individual features. This is due to the inherent characteristics of Malayalam scripts and low allographic variation among different writers. The features identified in all the three schemes were analysed to find their influence on the stability, consistency and performance of the system.

Elimination of redundant character was required only in grapheme based and character based schemes. This module used three features

(randomized selection, height-width ratio & curvature and geometrical feature) and three distance measures (Euclidean, chi-square and Manhattan) for this purpose. Through regression analysis it was shown that all the three features had significance. Experimental analysis proved that geometric features play the most significant role in the elimination of redundant characters among others. Also it was observed that Chi-square was much superior to other distance measures as the number of writers increased.

Stability of features and classifiers are done by analyzing the performance at different reference amount of text (zones) varying from one word - two words; one line - two lines - three lines - four lines; one paragraph; full page etc. on the documents of the total 280 writers. From the experimental observations, we can understand that the stability of the combined features outperforms the individual features. In the case of classifiers it is observed that performance of the Naive Bayes classifier was too low when compared to other classifiers.

Evaluation of the influence of the consistency of the features on identification rate across the number of writers was done in terms of coefficient of variation. In grapheme based scheme, curvature and angle pair features maintained consistency when compared with others. In character based scheme all features except the slant of the character and distance feature maintained consistency. In the case of image processing techniques (document level) only SIFT feature maintained its consistency.

The performance of classification models was done using the statistical measure namely sensitivity, specificity, precision and F-measure. It was evident that character based approach was most suitable for writer identification of Malayalam documents. The experiments had revealed that the curvature and directional features with geometric properties of a character are the prominent decisive features for writer identification of

handwritten Malayalam documents.

The prominent decisive features thus identified were utilized to develop an online character recognition system for recognizing the Malayalam characters and Grantha script. The different classifiers were used of which k-NN classifier showed good results.

8.2 Future Directions

- **Extending the work for behavioral prediction analysis of person.** This study helps in the identification of the uniqueness of one's writing style. This can be extended to the next level where the analysis of the handwriting of one at different situations and correlated to his behavior with in different moods. Thus the writer's behavior could be predicted based on his writing style.
- **Extending the work for offline and online signature verification.** Here we were able to extract the writer dependent feature from handwritten Malayalam documents. The same method could be extended to signature analysis and verification. This has large scale applications in various fields.
- **Extending the work for offline historic document analysis.** Here efforts were made to analyse Grantha script and convert it to more readable Malayalam script online. Most of the historical documents of Kerala are written in Grantha script which could not be read as the script outdated. But these documents are invaluable. Hence an offline method can be devised whereby the Grantha documents can be regenerated in modern languages.

The findings of this thesis can form a strong base for arenas described above.

References

[1] Srihari,S., Cha,S., Arora,H., and Lee,S. *"Individuality of Handwriting",* vol. 47, no. 4, July 2002, pp. 1-17.

[2] Sreeraj,M., Sumam Mary Idicula , *"OnLine Handwritten Character Recognition using Kohonen Networks",* Proceedings of the IEEE 2009 World Congress on Nature and Biologically Inspired Computing (NABIC '09),2009 pp 1425 - 1430.

[3] Bulacu,M.,Schomaker,L.,Brink,A., *"Text independent writer identification and verification on offline Arabic handwriting",* in Ninth Conference on Document Analysis and Recognition(ICDAR),2007.

[4] Fornes,A.,Llados, J.,Sanchez, G.,Bunke, H., *"Writer Identification in Old Handwritten Music Scores",* In 8th IAPR Workshop on Document Analysis Systems,2008, 347353.

[5] Sas, J , *"Handwriting Recognition Accuracy Improvement by Author Identification",* In L. Rutkowski et al. (eds.), ICAISC 2006, LNAI 4029, 682–691.Springer, Heidelberg

[6] Chaudhry, R., Pant, S. K. , *"Identification of authorship using lateral palm print a new concept",* J.Forensic Science International,2004 volume (141), 49–57

[7] Schomaker, L , *"Advances in Writer Identification and Verification",* In: 9th International Conference on Document Analysis and Recognition (ICDAR07),2007, volume (2), 1268–1273.

[8] Plamondon, R., Lorette, G. , *"Automatic Signature Verification and Writer Identification The State of the Art",* Pattern Recognition,1989 vol. 22, no. 2, pp. 107-131

[9] Leclerc ,F., Plamondon,R. , *"Automatic signature verification: the state of the art 1989-1993",* Int. J. Pattern Recognition Artific. Intell. 1994, 8,pp 643-659

[10] Gupta, S. , *"Automatic Person Identification and Verification using Online Handwriting",* Master Thesis. International Institute of Information Technology Hyderabad, India

[11] Schlapbach, A., Marcus, L., Bunke, H , *"A writer identification system for online whiteboard data,",* Pattern Recognition Journal 41,2008,2382123897.

[12] Arazi, B. , *"Handwriting identification by means of run length measurements",* IEEE Transactions of System, Man and Cybernatics,1977, vol. 7, pp. 878881.

[13] Arazi, B , *"Automatic handwriting identification based on the external properties of the samples",* IEEE Transactions of System, Man and Cybernatics,1983 vol. 13, pp. 635642

[14] Zimmerman, K., Varady, M. , *"Hand writer identification from one bit quantized pressure patterns",* Pattern Recognition,1985, vol. 18, no. 1, pp. 6372.

[15] He,Z., You,X., Tang ,Y.Y., *"Writer identification using global wavelet based features Neuro computing",* 2008,pp. 18321841.

[16] Bulacu, M., Schomaker, L., Brink, A. , *"Text independent writer identification and verification on offline Arabic handwriting",* in: Ninth Conference on Document Analysis and Recognition(ICDAR),2007.

[17] Helli, B., Moghaddam, M.E , *" A text independent Persian writer identification system using LCS based classifier",* in: IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2008.

[18] Tan, T. N , *"Texture feature extraction via visual cortical channel modeling",* in International Conference of Pattern Recognition,1992 vol. 3, pp. 607610.

[19] Said, H., Peake, G., Tan, T., Baker, K. , *"Writer Identification from Non Uniformly Skewed Handwriting Images",* Proc. Ninth British Machine Vision Conf.,1998, pp. 478-487.

[20] Zhenyu, T. Y., He, Xinge, Y , *"A contourlet based method for writer identification",* International Conference on Systems, Man and Cybernetics, vol. 1,2005 pp. 364368.

[21] He, J. Y. Z., Fang, B., You, X, *"A novel method for off line handwriting based writer identification",* in International Conference on Document Analysis and Recognition,2005, pp. 242256.

[22] Bulacu, M., Schomaker, L., Vuurpijl, L. , *"Writer identification using edge based directional features",* in: Seventh International Conference on Document Analysis and Recognition (ICDAR),2003.

[23] Bulacu, M. Schomaker, L , *"Writer Style from Oriented Edge Fragments",* Proc. 10th Intl Conf. Computer Analysis of Images and Patterns,2003, pp. 460-469.

[24] Bensefia, A., Paquet, T., Heutte, L , *"A Writer Identification and Verification System",* Pattern Recognition Letters,2005 vol. 26, no. 10, pp. 2080-2092

[25] Schomaker, L. Bulacu , *"Automatic writer identification using connected component contours and edge-based features of uppercase western script",* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004,vol. 26, pp. 787798.

[26] Seropian, A., Grimaldi, M., Vincent, N , *"Writer identification based on the fractal construction of a reference base",* in International Conference of Document Analysis and Recognition,2003, pp. 11631167.

[27] Hertel, C., Bunke, H , *"A Set of Novel Features for Writer Identification",* Proc. Fourth Intl Conf. Audio and Video-Based Biometric Person Authentication, pp. 679-687

[28] Marti, U.V., Messerli, R., Bunke, H , *"Writer Identification Using Text Line Based Features",* Proc. Sixth Intl Conf. Document Analysis and Recognition (ICDAR),2001,pp. 101-105.

[29] Tan,G.X., Viard-Gaudin,C., Kot,A.C., *"Automatic writer identification framework for online handwritten documents using character prototypes",* Pattern Recognition, 42 ,2009, pp. 33133323

[30] Ram, S.S., Moghaddam, M.E , *"Text independent Persian writer identification using fuzzy clustering approach",* in: International Con- ference on Information Management and Engineering (ICIME), Malaysia,2009

[31] Zhang, B., Srihari, S , *"Analysis of Handwritten Individuality Using Word Features ",* Proc. Seventh Intl Conf. Document Analysis and Recognition (ICDAR),2003, pp. 1142-1146.

[32] Tomai, C., Zhang, B., Srihari, S , *"Discriminatory Power of Handwritten Words for Writer Recognition",* Proc. 17th Intl Conf. Pattern Recognition,2004, pp. 638-641

[33] Zois, E., Anastassopoulos, V , *"Morphological Waveform Coding for Writer Identification",* Pattern Recognition,2000, vol. 33, no. 3, pp. 385-398.

[34] Al-Maadeed, S., Mohammed, E., AlKassis, D., Al-Muslih, F , *"Writer identification using edge based directional probability distribution features for Arabic words",* in: IEEE/ACS International Conference on Computer Systems and Applications (AICCSA),2008 582590.

[35] Zhang, B., Srihari, S., Lee, S , *"Individuality of Handwritten Characters",* Proc. Seventh Intl Conf. Document Analysis and Recognition (ICDAR),2003, pp. 1086-1090.

[36] Srihari, S., Tomai, C., Zhang, B., Lee, S. , *"Individuality of Numerals",* Proc. Seventh Intl Conf. Document Analysis and Recognition (ICDAR),2003, pp. 1096-1100.

[37] Leeham, G., Chachra, S , *"Writer identification using innovative binarised features of handwriting numerals",* in: Seventh International Conference on Document Analysis and Recognition (ICDAR),2003.

[38] Pervouchine, V., Leedham, G , *"Extraction and analysis of forensic document examiner features used for writer identification",* Pattern Recognition Journal 40,2007, 10041013

[39] Sutanto, P.,Leedham, G., Pervouchine, V , *"Study of the consistency of some discriminatory features used by document examiners in the analysis of handwritten letter a",* in International Conference on Document Analysis and Recognition,2003 pp. 10911095

[40] Pervouchine, V., Leedham, G, *"Extraction and analysis of document examiner features from vector skeletons of grapheme th,",* in Document Analysis Systems,2006,pp. 196207.

[41] Xianliang, X. D., Wang, Liu, H , *"Writer identification using directional element features and linear transform",* in International Conference on Document Analysis and Recognition,2003.

[42] Soleymani Baghshah, M., Bagheri Shouraki, S. Kasaei, S , *"A novel fuzzy classifier using fuzzy LVQ to recognize online Persian handwriting",* in: Second IEEE Conference on Information and Communication Technology (ICTTA),2006.

[43] Schlapbach, A., Bunke, H , *"A writer identification and verification system using HMM based recognizers",* Pattern Analysis Application (Springer),2007,10,3343, doi:10.1007/s10044-006-0047-5.

[44] Schlapbach, A., Bunke, H , *"Writer identification using an HMM based hand writing recognition system: to normalize the input or not?",* In: 12th Conference of the International Graphonomics Society, Salerno, Italy, June 2629, 2005, pp.138142

[45] Seiichiro Hangai, S. Y., Hamamoto, T. , *"Online signature verification based on altitude and direction of pen movement, ",* vol. 1, 2000,pp. 489492

[46] Tsai, L. M. Y , *"Online writer identification using the point distribution model",* in International Conference on System, Man and Cybernetics, vol. 2,2005, pp. 12641268.

[47] Hiroshi Kameya, S.M., Oka, R , *"Figure based writer verification by matching between an arbitrary part of registered sequence and an input sequence extracted from on line handwritten figures",* in International Conference on Document Analysis and Recognition,2003.

[48] Chapran, J , *".Biometric writer identification: feature analysis and classification",* International Journal of Pattern Recognition and Artificial Intelligence 20(4),2006, 483503.

[49] Chapran, J., Fairhurst, M.C , *"Biometric writer identification based on the interdependency between static and dynamic features of handwriting",* in: Proceedings of the 10th International Workshop on Frontiers in Handwriting Recognition, 2006, pp. 505510.

[50] Namboodiri, A.M., Gupta, S , *"Text independent writer identification from online handwriting",* in Proceedings of 10th International Workshop on Frontiers in Handwriting Recognition, (La Baule, Centre de Congreee Atlantia, France),2006, pp. 2326.

[51] Jin, W., Wang, Y., Tan, T. , *"Text independent writer identification based on fusion of dynamic and static features",* in International Workshop Biometric Recognition Systems, 2005,p. 197.

[52] Nakamura, Y. Kidode, M , *"Individuality analysis of online kanji handwriting",* in International Conference on Document Analysis and Recognition,2005.

[53] Said, H., Tan, T., Baker, K. , *"Personal Identification Based on Handwriting",* Pattern Recognition, vol. 33, no. 1,2000, pp. 149-160

[54] Tan, T , *"Rotation Invariant Texture Features and Their Use in Automatic Script Identification",* IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 20, no. 7,1998, pp. 751-756.

[55] Zhu, Y., Tan, T., Wang, Y. , *"Font Recognition Based on Global Texture Analysis",* IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 23, no. 10,2001, pp. 1192-1200.

[56] Marti, U.V., Bunke, H. , *"The IAM Database: An English Sentence Database for Offline Handwriting Recognition",* Intl J. Document Analysis and Recognition, vol. 5, no. 1,2002, pp. 39-46.

[57] Srihari, S., Beal, M., Bandi, K., Shah, V., Krishnamurthy, P., *"A Statistical Model for Writer Verification",* Proc. Eighth Intl Conf. Document Analysis and Recognition (ICDAR),2005, pp. 1105-1109.

[58] Favata, J., Srikantan, G, *"A Multiple Feature/Resolution Approach to Hand printed Digit and Character Recognition",* Intl J. Imaging Systems and Technology, vol. 7,1996, pp. 304-311.

[59] Bensefia, A., Paquet, T., Heutte, L. , *"Handwritten Document Analysis for Automatic Writer Recognition",* Electronic Letters on Computer Vision and Image Analysis,2005, vol. 5, no. 2, pp. 72-86.

[60] Bensefia, A., Nosary, A., Paquet, T., Heutte, L. , *"Writer Identification by Writers Invariants" ,* Proc. Eighth Intl Workshop Frontiers in Handwriting Recognition" 2002,pp. 274-279.

[61] Bensefia, A., Paquet, T., Heutte, L. , *"Information Retrieval Based Writer Identification",* Proc. Seventh Intl Conf. Document Analysis and Recognition (ICDAR), pp. 946-950.

[62] Schlapbach, A., Kilchherr, V., Bunke, H. , *"Improving Writer Identification by Means of Feature Selection and Extraction",* Proc. Eighth Intl Conf. Document Analysis and Recognition (ICDAR),2005, pp. 131-135.

[63] Vander Maaten, L., Postma, E. , *"Improving automatic writer identification",* in:17thBelgium-Netherland Conference on Artificial Intelligence.2005

[64] Pervouchine, V., Leedham, G., Melikhov, K., *"Handwritten character skeletonisation for forensic document analysis",* in: ACM Symposium on Applied Computing,2005.

[65] Schomaker, L., Franke, K., Bulacu, M. , *""Using codebooks of fragmented connected component contours in forensic and historic writer identification,* Pattern RecognitionLetter28,2007, 719727.

[66] Schomaker, M.B.L. , *"Analysis of texture and connected component contours for the automatic identification of writers",* in: 16th BelgiumNetherland Conference on Artificial Intelligence(BNAIC),2004.

[67] Bulacu, M., Schomaker, L. , *"Combining multiple features for text independent writer identification and verification",* in:10th international Workshop on Frontiers in Handwriting Recognition(IWFHR),2006.

[68] Li, B., Sun, Z., Tan, T.N. , *" Hierarchical Shape Primitive Features for Online Text independent Writer Identification",* Proc. of 2th ICB,2007, pages 201210

[69] He, Z., You, X., Tang, Y.Y. , *"Writer identification of Chinese handwriting documents using hidden Markov tree model",* Pattern Recognition Journal 41, 129513072008-06-15,2008.

[70] Yan, Y., Chen, Q., Deng, W., Yuan, F. , *"Chinese Handwriting Identification Based on Stable Spectral Feature of Texture Images",* International Journal of Intelligent Engineering and Systems,2009, Vol.2, No.1.

[71] Bulacu, M., Schomaker, L., *"Text-independent writer identification and verification using textural and allographic features",* IEEE Transactions on Pattern Analysis and Machine Intelligence(PAMI)29(4) 701717 Special Issue Biometrics: Progress and Directions,2007.

[72] Al-Dmour, A., Zitar, R.A. , *"Arabic writer identification based on hybrid spectral statistical measures",* Journal of Experimental and Theoretical Artificial Intelligence,2007, 19(4) 307332.

[73] Feddaoui, N., Hamrouni, K. , *" Personal Identification based on Texture Analysis of Arabic Handwriting Text",* In: IEEE International Conference on Information and Communications Technologies (ICTTA06),2007, vol. (1), 1302-1307.

[74] Gazzah, S., Ben Amara, N. E., *"Arabic Handwriting Texture Analysis for Writer Identification using the DWT lifting Scheme",* In: 9th International Conference on Document Analysis and Recognition (ICDAR07), vol. (2),2007, 1133-1137

[75] Abdi, M. N., Khemakhem, M., Ben-Abdallah, H. , *"A Novel Approach for Off Line Arabic Writer Identification Based on Stroke Feature Combination.",* In: 24th IEEE International Symposium on Computer and Information Sciences, (ISCIS09),2009.

[76] Shahabi, F., Rahmati, M. , *"Comparison of Gabor based features for writer identification of Farsi/Arabic handwriting",* in: 10th International Workshop on Frontiers in Handwritten Recognition (IWFHR),2006.

[77] Helli, B., Moghaddam, M.E., *"Persian writer identification using extended Gabor filter",* in: International Conference on Image Analysis and Recognition (ICIAR),2008.

[78] Rafiee, A., Motavalli, H. , *"Offline Writer Recognition for Farsi text.",* In: 6th Mexican International Conference on Artificial Intelligence (MICAI07), Special Session, 193–197,2007.

[79] Helli, B., Moghaddam, M.E., *"A writer identification method based on XGabor and LCS",* IEICE Electronics Express 6 (10),2009.

[80] Ram, S.S., Moghaddam, M.E. , *"A Persian writer identification method based on gradient features and neural networks",* in: Second International Conference on Image and Signal Processing (CISP), China,2009.

[81] Srihari, S. N., Cha, S.-H., Lee, S. , *"Establishing Handwriting Individuality Using Pattern Recognition Techniques",* in Proceedings of the Sixth International Conference on Document Analysis and Recognition, pp. 1195-1204,2001.

[82] Long Zuo, Yunhong Wang, Tieniu Tan , *".Personal Handwriting Identification Based on PCA",* Proceedings of SPIE Second International Conference on Image and Graphics, pp 766-771,2002.

[83] Thumwarin, P., Matsuura, T. , *"On-line writer recognition for Thai based on velocity of barycenter of penpoint movement",* in: International Conference on Image Processing, ICIP '04, vol. 2,2004, pp. 889892.

[84] Schlapbach, A., Bunke, H. , *"Using HMM based recognizers for writer identification and verification",* in: ProceedingsInternational Workshop on Frontiers in Handwriting Recognition, IWFHR, Tokyo,2004, pp. 167172.

[85] Chan, S. K., Viard-Gaudin, C., Tay, Y. H. , *"Online writer identification using character prototypes distributions",* in Proceedings of SPIE - The International Society for Optical Engineering.,2008.

[86] Schomaker, L., Bulacu, M., Franke, K. , *"Automatic Writer Identification Using Fragmented Connected Component Contours",* Proc. Ninth Intl Workshop Frontiers in Handwriting Recognition (IWFHR),2004, pp. 185-190.

[87] Bulacu, M., Schomaker, L , *"A Comparison of Clustering Methods for Writer Identification and Verification",* Proc. Eighth Intl Conf. Document Analysis and Recognition, vol. II,2005, pp. 1275-1279.

[88] Niels, R., Gootjen, F. Vuurpijl, L. , *"Writer Identification through Information Retrieval: The Allograph Weight Vector",* in International Conference on Frontiers in Handwriting Recognition,2008, pp. 481-486.

[89] Peake, G.S, Tan,T.N. , *"Script and language identification from document images",* In: Proc. of the British Machine Vision Conference (BMVC97), Vol. 2, pp. 169-184, 1997

[90] Siddiqi,I., Vincent,N., *"Writer identification in handwritten documents, In: ICDAR 07",* Proceedings of the Ninth International Conference on Document Analysis and Recognition, vol. 1, pp. 108-112, 2007

[91] Blumenstein,M., Liu,X,Y. and Verma B. , *"An investigation of the modified direction feature for cursive character recognition",* J. Pattern Recognition, vol. 40, (2), pp. 376-388, 2007

[92] Yamada H and Nakano Y., *"Cursive handwritten word recognition using multiple segmentation determined by contour analysis",* J. IEICE Transactions on Information and Systems, E79-D, pp. 464-470, 1996

[93] Blumenstein M, Verma B and Basli H , *"A novel feature extraction technique for the recognition of segmented handwritten characters",* In: ICDAR 03: Proceedings of the Seventh International Conference on Document Analysis and Recognition, pp. 137-141, 2003

[94] Kimura F, Kayahara N, Miyake Y and Shridhar M. , *"Machine and human recognition of segmented characters from handwritten words",* In: ICDAR 97: Proceedings of the Fourth International Conference on Document Analysis and Recognition, pp. 866-869, 1997

[95] Dehkordi M E, Sherkat N and Allen T. , *"Handwriting style classification",* Int. J. on Document Analysis and Recognition, 6, pp. 55-74,2003

[96] Chang F, Chou C H, Lin C C and Chen C J. , *"A prototype classification method and its application to handwritten character recognition",* In: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, 2004

[97] Solihin,Y., *"A toolset of image processing algorithms for forensic document examination. Technical Report",* School of Applied Science, Nanyang Technological University, Singapore, 1997.

[98] Rehman,A., Mohammad,D., Sulong,G., *"Simple and Effective Technques for Core region Detection and Slant Correction in Script Recognition",* Proceedings of IEEE, International Conference on Signal and Image Processing, 2009.

[99] Kavallieratou,E., Fakotakis,N.,Kokkinakis,G., *"Slant estimation algorithm for OCR systems",* Pattern Recognition, Vol.34, no.12, Dec.2001, pp.2515-2522

[100] Kherallah, M., Haddad, L., Alimi, A.M., Mitiche,A. , *"Online handwritten digit recognition based on trajectory and velocity modeling",* Pattern Recognition Letters 29(5),580594 ,2008.

[101] Garain,U., Paquet,T., *"Off-Line Multi Script Writer Identification Using AR Coefficients",* In Proc.10th International Conf. on Document Analysis and Recognition, 2009, pp. 991-995.

[102] Wong,K., Casey,R., Wahl,F., *"Document analysis system",* IBM J. Research and Development, 26(6),1982.647-656

[103] Nagy,G., Seth, Viswanathan,M., *"A prototype document image analysis system for technical journals",* IEEE Comput. 25 (July 1992), pp. 1022

[104] Lee,H.Y.,Kim,H.,Lee,H.K., *"Robust image watermarking using local invariant features",* Opt. Eng., 45 (3) (2006) 037002(1-11)

[105] Enrico, G., Manuele, B., Anderea, L., Massimo, T , *"On the use of SIFT features for face authentication",* In the proceedings of the conference on Computer Vision and Pattern Recognition Workshop, pp. 91110, 2006

[106] Schwarz, P , *"Recognition of Graffit",* BS Thesis, The University of Western Australia, 2006.

[107] Dlagnekov, L, *"Video-based Car Surveillance: Licence plate, Make and Model Recognition",* , MSc Thesis, University of California, San Diego, 2005.

[108] Park, U., Pankanti, S., Jain, A.K , *"Fingerprint Verification using SIFT Features",* In: Proc. of SPIE Defense and Security Symposium, Orlando, Florida (2008).

[109] Liang Du, Xinge You, Huihui Xu, Zhifan Gao , *"Yuan Yan Tang: Wavelet Domain Local Binary Pattern Features For Writer Identification",* ICPR 2010: 3691-3694

[110] Lowe,D. , *"Distinctive Image features from Scale- invariant Keypoints",* J. Computer Vision., vol. 60, no. 2, pp. 91110, 2004

[111] Gonzalez,R.,Woods,R. , *"Digital Image Processing, Second edition",* Pearson Education, India, 2006.

[112] Rejean Plamondon, N. Srihari , *"Online and off-line handwriting recognition: a comprehensive survey",* , J.IEEE transactions on pattern analysis and machine intelligence, vol. 22(2000).

[113] Swethalakshmi,H., *"Online Handwritten Character recognition for Devanagari and Tamil Scripts using Support Vector Machines",* Masters thesis, Indian Institute of Technology, Madras, India, 2007.

[114] Swethalakshmi,H.,Jayaraman,A.,Chakravarthy,V.S.,Sekhar,C.C., *"Online Handwritten Character Recognition of Devanagari and Telugu Characters using Support Vector Machines",* 10th Int. Frontiers in Handwriting Recognition Workshop (IWFHR 2006), La Baule, France, 2006.

[115] Jayaraman,A.,Sekhar,C.C.,Chakravarthy, V.S., *" Modular Approach to Recognition of Strokes in Telugu Script",* 9th Int. Document Analysis and Recognition Conference (ICDAR 2007), Curitiba, Brazil, 2007.

[116] Aparna,K.H.,Subramanian,V.,Kasirajan,M., Prakash,G.V., Chakravarthy,V.S.,Madhvanath,S., *"Online Handwriting Recognition for Tamil, 9th Int. Frontiers in Handwriting ",* Recognition Workshop (IWFHR 2004), Tokyo, Japan, 2004.

[117] Babu,V.J., Prashanth,L.,Sharma,R.R., Rao,G.V.P., Bharath,A., *"HMM Based Online Handwriting Recognition System for Telugu Symbols",* Proc. 9th Int. Document Analysis and Recognition Conference (ICDAR 2007), Curitiba, Brazil, 2007.

[118] Joshi,N.,Sita,G., Ramakrishnan,A.G.,Deepu,V.,Madhvanath,S. *"Machine Recognition of Online Handwritten Devanagari Characters",* 8th Int. Document Analysis and Recognition Conference (ICDAR 2005), Seoul, Korea, 2005.

[119] Deepu, S. Madhvanath , *"Genetically Evolved Transformations for Rescaling Online Handwritten Characters",* IEEE India Annual Conference (INDICON 2004), Kharagpur, India, 2004.

[120] Toseli,A. H.,Pastor,M., Vidal, E. *"Online Handwriting Recognition System for Tamil Handwritten Characters",* J. Pattern Recognition and Image Analysis, Berlin / Heidelberg (2007) 370-377.

[121] Prasanth,L.,Babu, V.J., Sharma, R.R., Rao,G.V.P., Dinesh, M. , *"Elastic Matching of Online Handwritten Tamil and Telugu Scripts using Local Features",* 9th Int. Document Analysis and Recognition Conference (ICDAR 2007), Curitiba, Brazil, 2007

[122] Sundaram,S., Ramakrishnan,A.G. , *"A Novel Hierarchical Classification Scheme for Online Tamil Character Recognition",* 9th Int. Document Analysis and Recognition Conference (ICDAR 2007), Curitiba, Brazil, 2007.

[123] Sundaresan,C. S., Keerthi, S. S., *"A Study of Representations for Pen based Handwriting Recognition of Tamil Characters",* 5th Int. Document Analysis and Recognition Conference (ICDAR 1999), Bangalore, India, 1999.

[124] Rao,P.V.S., Ajitha,T.M., *"Telugu Script Recognition: A Feature based Approach",* 3rd Int. Document Analysis and Recognition Conference (ICDAR 1995), Montreal, Canada, 1995.

[125] Connell,S.D.,Sinha,R.M.K.,Jain,A.K., *"Recognition of Unconstrained Online Devanagari Characters",* 15th Int. Pattern Recognition Conference (ICPR 2000), Barcelona, Spain, 2000.

[126] Joshi,N.,Sita,G., Ramakrishnan, A.G.,Madhvanath,S. *"Comparison of Elastic Matching Algorithms for Online Tamil Handwritten Character*

Recognition", 9th Int. Frontiers in Handwriting Recognition Workshop (IWFHR 2004), Tokyo, Japan, 2004.

[127] Ranade, M. Ranade , *", Devanagari Pen-written Character Recognition",* 9th Int. Advanced Computing and Communications Conference (ADCOM 2001), India, 2001

[128] Kunte,R.S.R, Samuel,R.D.S., *" Online Character Recognition System for Handwritten Characters/Script with Bilingual Facility Employing Neural Classifiers and Wavelet Features ",* , Int. Knowledge based Computer Systems Conference (KBCS 2000), Mumbai, India, 2000.

[129] Bhattacharya,U.,Gupta,B.K.,Parui,S.K., *", Direction Code based Features for Recognition of Online Handwritten Characters of Bangla,",* 9th Int. Document Analysis and Recognition Conference (ICDAR 2007), Curitiba, Brazil, 2007

[130] Jayababu,G., Sumam Mary Idicula , *"Hand Written Malayalam Character Recognition an Approach Based on Pen Movement",* Int. Knowledge Management Conference, Malaysia , 2004.

[131] Deepu,V.,Madhvanath,S.,Ramakrishnan, A.G. , *"Principal Component Analysis for Online Handwritten Character Recognition",* 17th Int. Pattern Recognition Conference (ICPR 2004), Cambridge, United Kingdom, 2004.

[132] Joshi,N.,Sita,G., Ramakrishnan,A.G.,Madhvanath,S., *" Tamil Handwriting Recognition Using Subspace and DTW Based Classifiers",* 11th Int. Neural Information Processing Conference (ICONIP 2004), Calcutta, India, 2004.

[133] `http://unipen.nici.ru.nl/unipen.def`

[134] Guyon,I., Schomaker,L.,Plamondon, R.,Liberman, M., Janet,S. , *", UNIPEN Project of Online Data Exchange and Recognizer Benchmarks",* International Conference on Pattern Recognition (ICPR 1994), Jerusalem, Israel 1994.

[135] Jaeger,S., Manke, S., Reichert,J., Waibel,A. *"Online handwriting recognition: the NPen++ recognizer",* J.Intl. Document Analysis and Recognition, March, vol.3, (2001), 169-180

[136] Manke,S.,Finke, M., Waibel, A. , *"Combining Bitmaps with Dynamic Writing Information for OnLine Handwriting Recognition",* . In proceedings of the 12th International Conference on Pattern Recognition, 1994, pp 596-598.

[137] Otsu,N. , *"A Threshold Selection Method from Gray-Level Histograms",* J.IEEE Transactions on Systems, Man, and Cybernetics, (1979).

[138] Muhammad Faisal Zafar, Dzulkifli Mohamad, Razib M , *"Othman Online Handwritten Character Recognition: An Implementation of Counter propagation Neural",* Proceedings of world academy of science, Engineering and Technology volume 10, 2005.

[139] Pastor,M.,Toselli,A.,Vidal, E., *"Writing Speed Normalization for Online Handwritten Text Recognition",* , Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), 2005

[140] Guyon,I.,Albrecht,P.,Le Cun,Y., Denker,J., Hubbard,W. , *"Design of a Neural Network Character Recognizer for a Touch Terminal",* ,J. Pattern Recognition,(1991) 105119

[141] Sreeraj,M., Sumam Mary Idicula, *"k-NN Based OnLine Handwritten Character Recognition System, ",* Integrated Intelligent Computing

(ICIIC),), 2010 First International Conference on , vol., no., pp.171-176, 5-7, 2010

[142] Shelke,S., Apte, S., *"A Multistage Handwritten Marathi Compound Character Recognition Scheme using Neural Networks and Wavelet Features",* J.Intl Signal Processing, Image Processing and Pattern Recognition, vol. 4 no. 1 (2011) 81-94.

[143] R. Gonzalez, and R. Woods, *"Digital Image Processing, Second edition",* Pearson Education, India, 2006.

[144] Cheriet,M., Kharma, N., C.-L. Liu, and C. Y. Suen, *"Character Recognition Systems A Guide for Students and Practitioners",* John Wiley and Sons, Inc, 2007.

[145] Neila Mezghani, Amar Mitiche, Mohamed Cheriet , *"Online recognition of handwritten Arabic characters using A Kohonen neural network",* Proceedings of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR02) IEEE, 2002

[146] Carpenter,G. A., Grossberg,S., Markuzon, N., Reynolds J. H., Rosen, D. B. , *"Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps",* J.IEEE Trans. on Neural Networks, 3(5) (1992) 698713.

[147] Carpenter,G. A., Grossberg,S., *"A self-organizing neural network for supervised learning, recognition, and prediction",* J.IEEE Communications Mag. (1992) 3849.

[148] Vakil-Baghmisheh M. T., Pavesic,N , *"fast simplified fuzzy ARTMAP network",* J.Neural Processing Letters (2003) 17: 273

[149] TRIER, O. D., JAIN, AND A. K., TAXT, T , *"Feature Extraction Methods For character Recognition : A survey",* Pattern Recognition, 29, 4, 641 662,1992.

[150] `http://tdil.mit.gov.in/pdf/Grantha/pdf_Unicode_proposal_Grantha_.pdf`

[151] `http://www.ancientscripts.com/sa_ws.html`

[152] `http://www.virtualvinodh.com/grantha-lipitva`

[153] B.V Dhandra and Mallikarjun Hangarge, *"Global and Local Features Based Handwritten Text Words and Numerals Script Identification",* ,International Conference on Computational Intelligence and Multimedia Applications, IEEE 471 475,2007

[154] Pulak Purkait, Rajesh Kumar and Bhabatosh Chanda, *"Writer Identification for Handwritten Telugu Documents using Directional Morphological Features",* , 12th International Conference on Frontiers in Handwriting Recognition, IEEE 658 663, 2010

[155] Utpal Garain and Thierry Paquet, *"Off-Line Multi-Script Writer Identification using AR Coefficients",* ,10th International Conference on Document Analysis and Recognition, IEEE 991 995, 2009

List of Publications

Papers in International Journals

1. Sreeraj.M and Sumam Mary Idicula. *"A Survey on Writer Identification Schemes"*, International Journal of Computer Applications 26(2):23-33, July 2011. Published by Foundation of Computer Science, New York, USA.

2. Sreeraj.M and Sumam Mary Idicula, *"Identifying Decisive Features for Distinctive Analysis of Writings in Malayalam"*, International Magazine on Advances in Computer Science and Telecommunications (IMACST), Association for Computer Science and Telecommunications (AKOMNAT TEL DOO), Vol.2 , No.1, pp. 13-20, May,2011.

3. Sreeraj.M and Sumam Mary Idicula, *"Writer Identification in Malayalam using Graphemes - A Decisive Evaluation"*, International Journal on Information Processing(IJIP), Vol.5, No.4 , pp. 45-53, December 2011

4. Sreeraj.M and Sumam Mary Idicula, *" An Online Character Recognition System to Convert Grantha Script to Malayalam"*, International Journal of Advanced Computer Science and Applications (IJACSA) (Accepted for Volume 3 Issue 7 July 2012)

Book Chapters

1. Sreeraj M. and Sumam Mary Idicula, *"A Novel Approach to Writer Identification in Malayalam using Graphemes"*, 5th International Conference on Information Processing, ICIP 2011, Bangalore, India, Communications in Computer and Information Science (CCIS), Springer-Verlag, Vol. 157, pp.646-651, August 5-7,2011

Papers in International Conferences

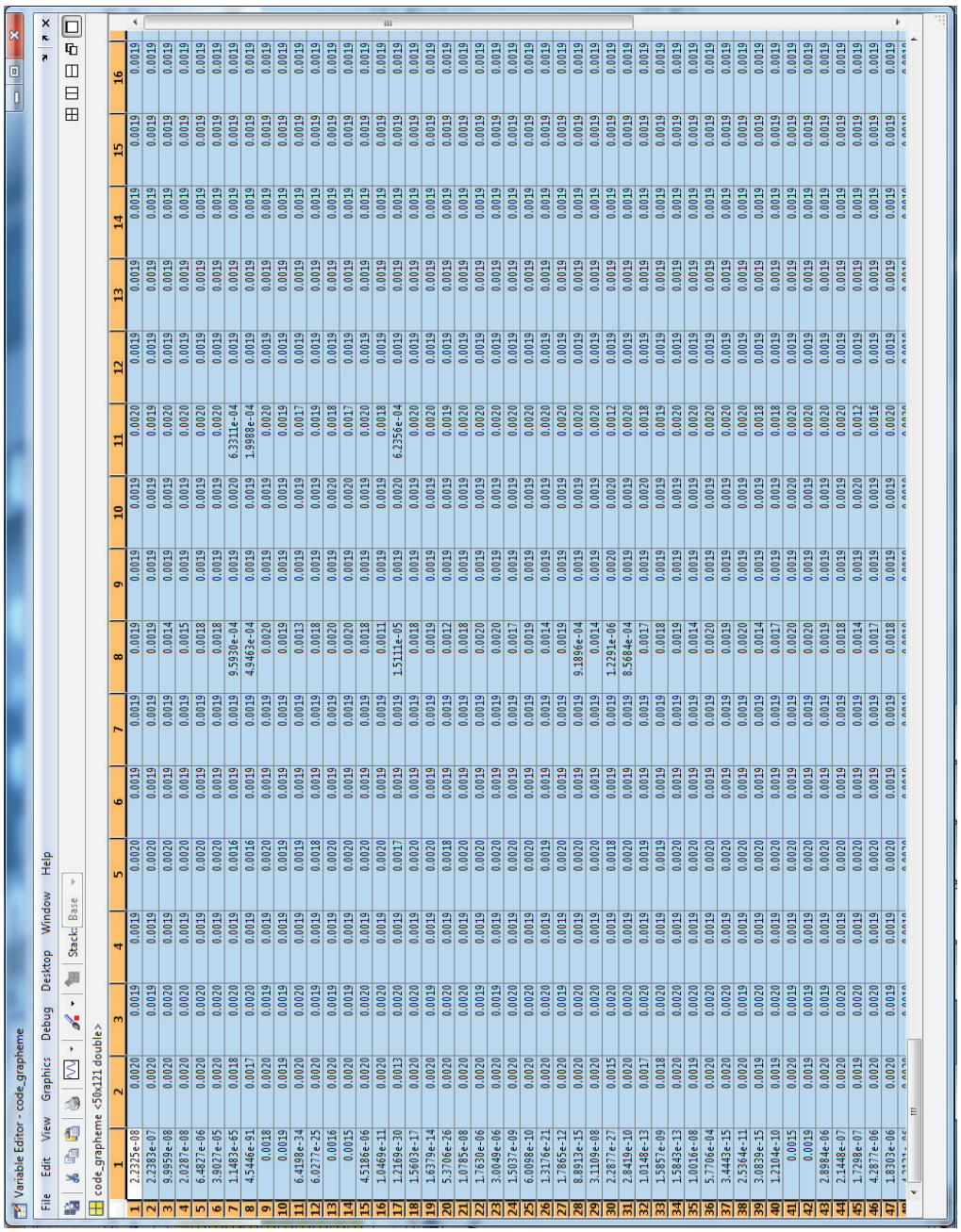
1. M. Sreeraj, Sumam Mary Idicula, *"k-NN Based On-Line Handwritten Character Recognition System"*, First International Conference on Integrated Intelligent Computing, iciic 2010 (Bangalore),IEEE Computer Society Press, pp.171-176, 2010.

2. M.Sreeraj, Sumam Mary Idicula, *"Exploiting time-domain and directional features for online handwriting recognition of Malayalam characters"*, International Conference on "Advances and Emerging Trends in Computing Technologies'(ICAET10), Chennai, June 21-24, 2010

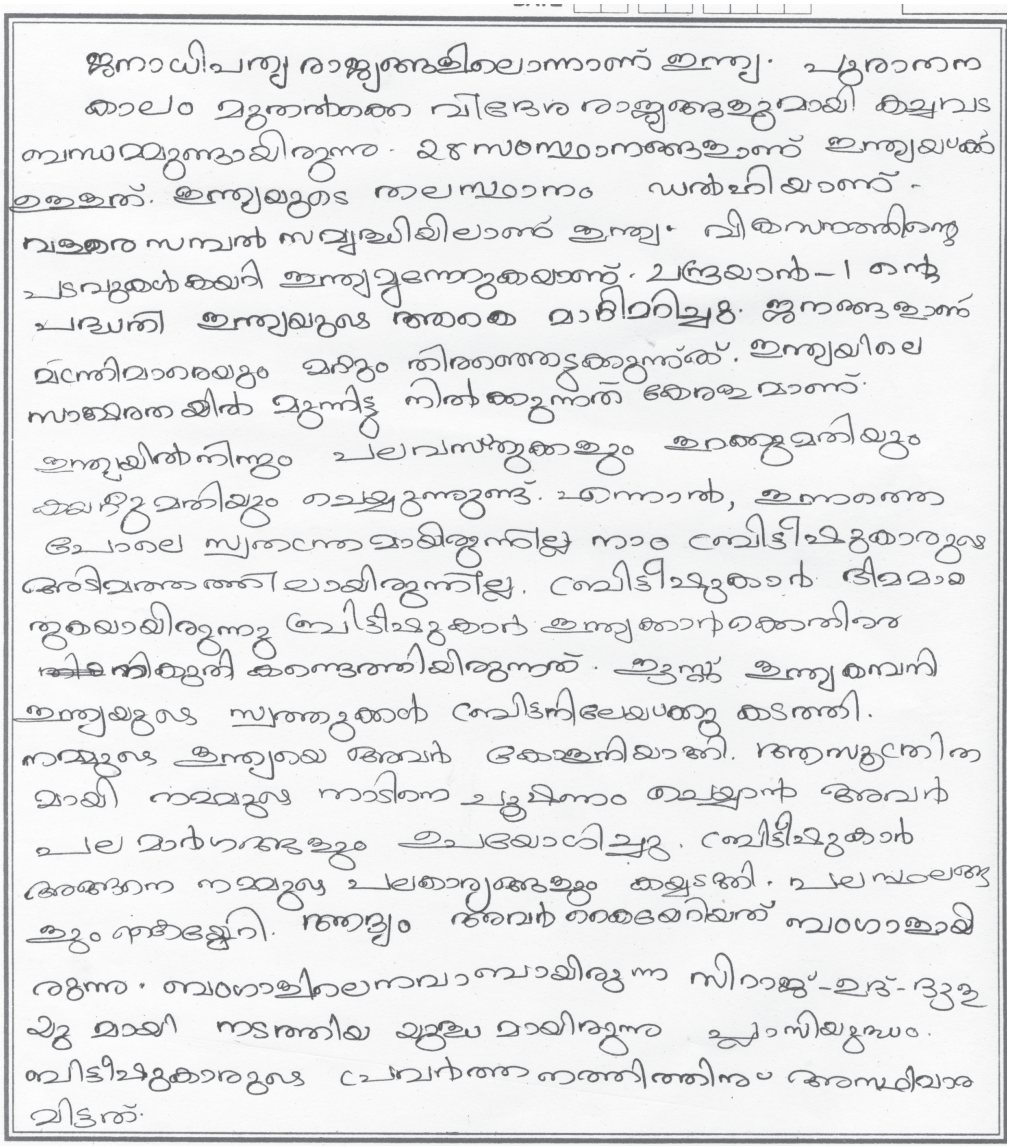
3. M. Sreeraj, Sumam Mary Idicula, *"On-Line Handwritten Character Recognition using Kohonen Networks"*, World Congress on Nature & Biologically Inspired Computing (NABIC '09), IEEE Computer Society Press, pp.1425 - 1430, 2009.

4. M. Sreeraj, Sumam Mary Idicula, *"The Effect of SIFT Features as Content Descriptors in the Context of Automatic Writer Identification in Malayalam Language"*, Fourth IEEE International Congress on Ultramodern Telecommunications and Data Security(ICUMT 2012), (communicated).

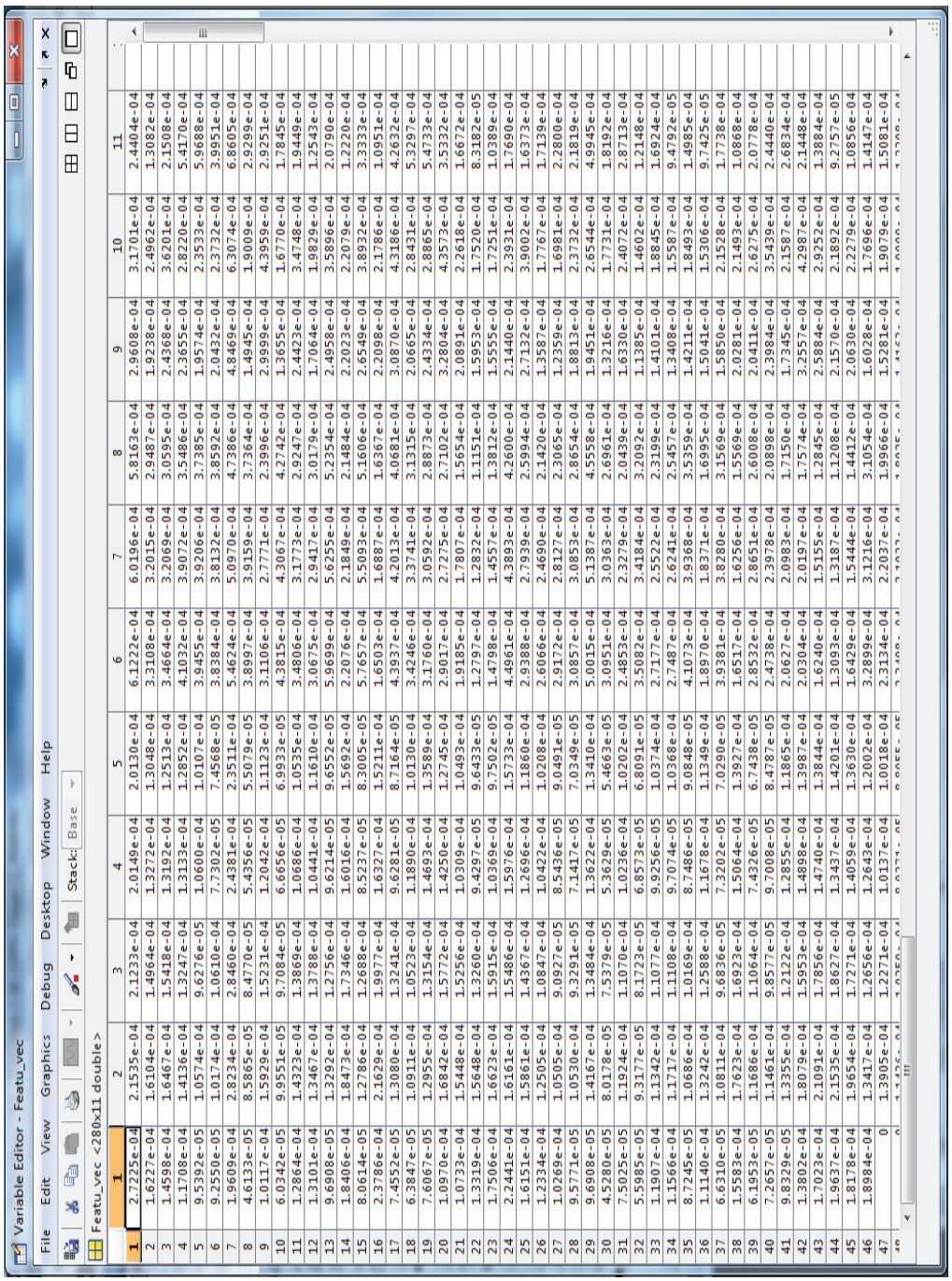
Appendix A Codebook Grapheme



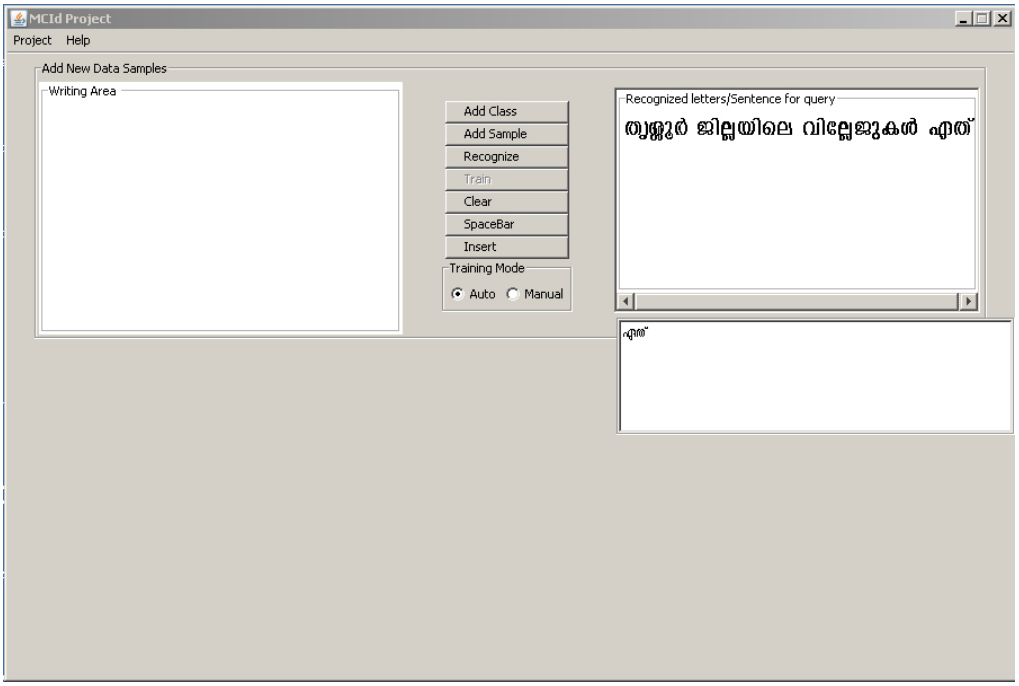
Appendix B Sample Handwriting



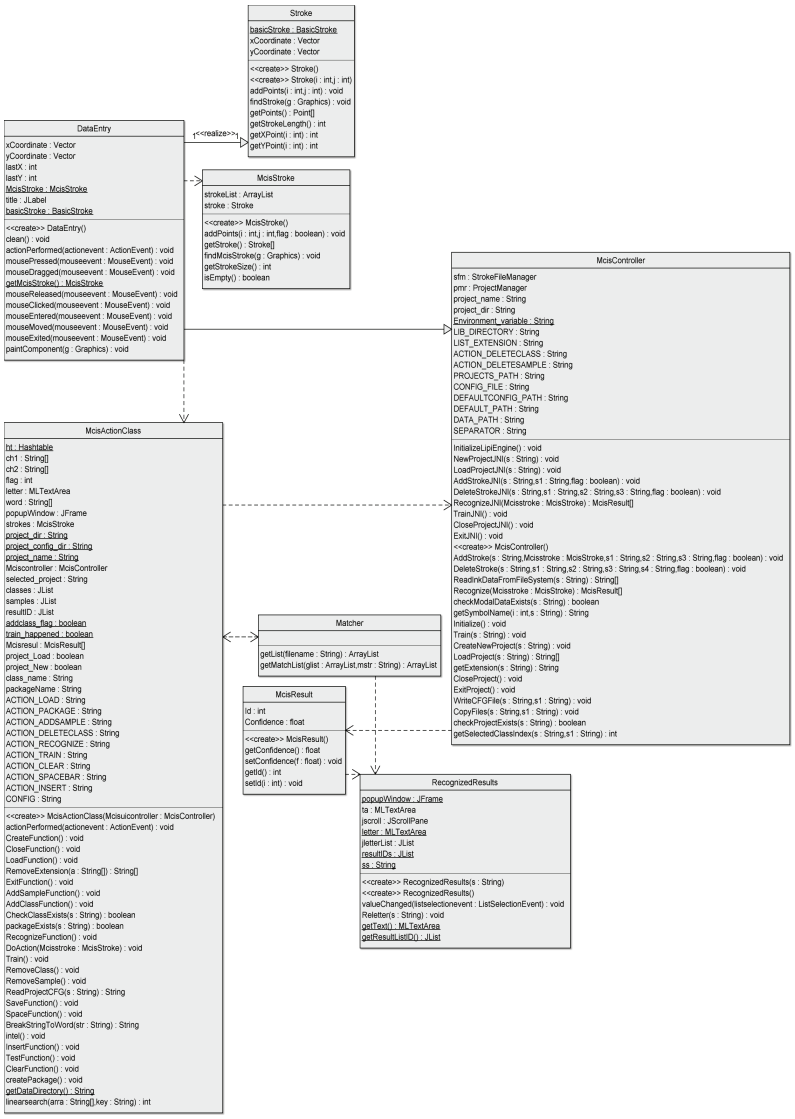
Appendix C Codebook Character



Appendix D Screenshot of Online Malayalam Character Recognition

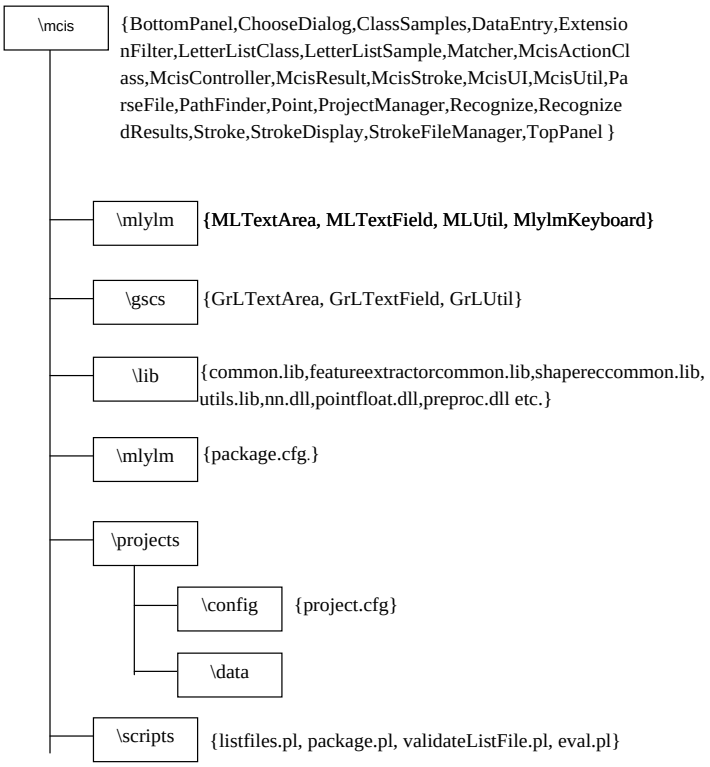


Appendix E Class Diagram of Application Framework



Appendix F Package Structure of the Application framework

Package Structure of the Application framework for online recognition of Malayalam and Grantha scripts

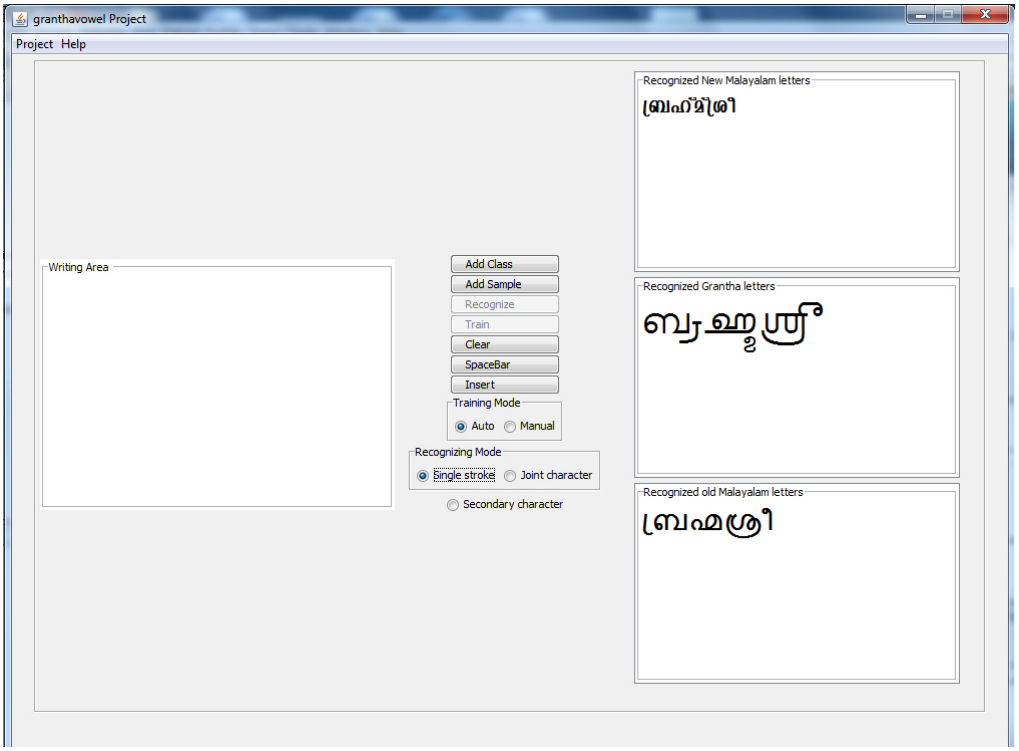


Appendix G Sample Unipen Format of Malayalam Character

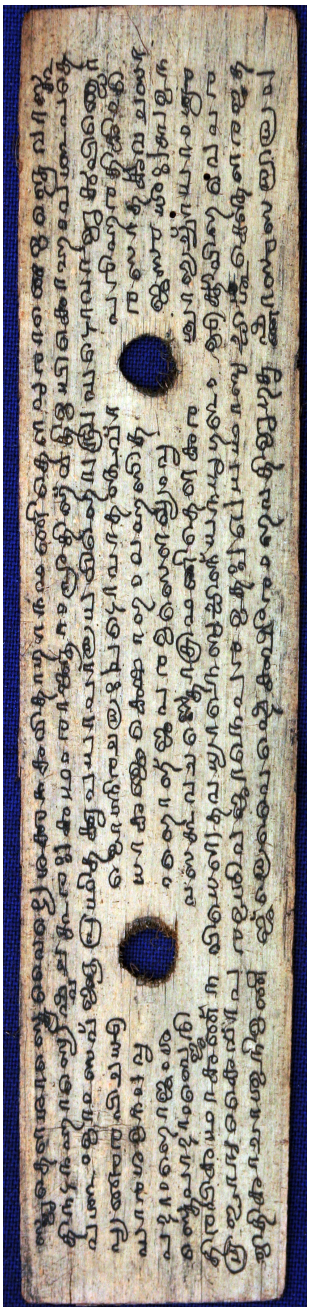
അ

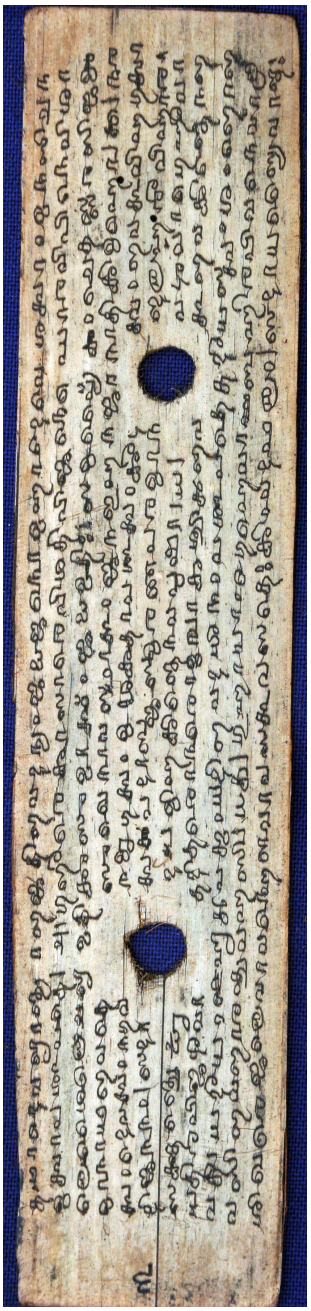
.VERSION 1.0
.HIERARCHY CHARACTER
.COORD X Y T
.SEGMENT CHARACTER
.X_POINTS_PER_INCH 100
.Y_POINTS_PER_INCH 100
.POINTS_PER_SECOND 75
.H_LINE 2760 3280
.V_LINE 2320 2840
.PEN_DOWN
41 121 0
41 121 0
41 120 0
41 119 0
40 119 0
39 117 0
39 115 0
38 113 0
38 111 0
37 109 0
36 107 0
36 104 0
35 102 0
35 99 0
35 96 0
35 93 0
35 90 0
35 87 0
35 84 0
35 82 0
35 80 0
35 78 0
35 76 0
37 74 0
38 72 0
39 70 0
40 69 0
42 66 0
43 66 0
45 64 0
46 63 0
47 63 0
49 63 0
50 62 0
51 62 0
52 62 0
54 62 0
55 62 0
57 63 0
58 64 0
59 64 0
59 65 0
59 67 0
59 68 0
59 69 0
59 71 0
59 74 0
57 77 0
56 79 0
55 80 0
53 83 0
52 84 0
50 85 0
49 86 0
48 87 0
49 87 0
50 87 0
52 87 0
53 87 0
55 88 0
56 89 0
57 89 0
59 91 0
60 91 0
61 93 0
62 94 0
62 95 0
63 97 0
64 99 0
64 100 0
63 101 0
62 103 0
62 104 0
61 105 0
60 106 0
60 107 0
59 107 0
58 108 0
57 108 0
56 108 0
55 108 0
55 107 0
54 107 0
54 105 0
53 104 0
52 102 0
51 100 0
51 97 0
50 95 0
50 94 0
50 92 0
50 90 0
50 89 0
50 88 0
51 86 0
52 84 0
54 82 0
56 80 0
59 78 0
62 76 0
64 76 0
66 75 0
68 74 0
70 74 0
73 74 0
76 74 0
79 75 0
81 77 0
83 78 0
85 80 0
88 82 0
90 84 0
91 87 0
92 89 0
93 91 0
94 94 0
95 97 0
95 100 0
95 102 0
95 105 0
95 106 0
95 108 0
95 109 0
95 110 0
94 110 0
94 109 0
94 108 0
94 106 0
94 104 0
94 102 0
96 98 0
97 96 0
98 93 0
100 90 0
101 87 0
104 85 0
106 82 0
109 79 0
111 76 0
115 73 0
118 71 0
120 69 0
124 68 0
127 67 0
129 66 0
130 66 0
132 65 0
133 65 0
134 65 0
135 66 0
136 67 0
138 69 0
140 71 0
141 74 0
143 77 0
145 81 0
146 83 0
146 86 0
148 89 0
148 93 0
148 98 0
148 101 0
148 104 0
146 108 0
144 111 0
142 113 0
140 116 0
136 118 0
133 119 0
130 120 0
128 121 0
126 121 0
123 121 0
122 120 0
120 119 0
120 118 0
119 116 0
118 115 0
118 113 0
117 110 0
117 108 0
117 106 0
117 103 0
118 101 0
119 99 0
121 98 0
123 96 0
125 95 0
129 94 0
132 93 0
135 92 0
137 91 0
138 91 0
139 91 0
140 91 0
.PEN_UP

Appendix H Grantha Recognition Screenshot



Appendix I Sample Manuscript





Appendix J A passage from the Book *Soundarya Lahari* written in Grantha script

Sample Grantha from ‘Soundarya Lahari’

சுவதாணிகா ஐஹவஊஶம்காஹமவத்ஜ்யவாஉா:பஉய

உததகுவெஉிநஸஉயாவர்ய்ம்உம்ஊகளாம்ஶ்ஜாகஶெதநவ்ர

ஸ்தூவம்தி வர்ய ஶிவஜதி ஶிவ: ஸவூஉம்மெளாவெத : ஸ

உாஶிவகுஉ வஶகாம்தள வித்யஸ்ஊாஊாதொஶ்ஶவஶ ப்ஜொநிஷ்ஷ

ஞ: ய யொக்தஉ ஹிஸியாதொஸ்ஸிம்ஹஶ ப்ஜொவஶகாம்தளஶிவ

ஸ்தூத: வணூவ்ருத்யதஸ்ஸிம்யள வஶ்யகு:கஶ்யவாஉிவக ஜதி

Equivalent Malayalam

മതതത്വ വെദിനസമയാവ്യാംചംദക്രളാംശ്ലൊകശ തേനപ്ര

സ്തൂവംതി വ്യാ ശിവഇതി ശിവഃ സർവമംഗ ളോപെത ഃ സ

ദാശിവതത്വമ് വശകാംതൗ വിത്യസ്മാദ്ധാ തോശ്ശിവ ശ ബ്ദോനിഷ്പ

ന്നഃ യ ഥോക്തമ് ഹിസിധാ തോസ്സിംഹശ ബ്ദോവശകാംതൗശിവ

സ്തൂതഃ വണർവ്യത്യയതസ്സിംധൗ പശ്യകഃകശ്യപാദിവത് ഇതി

Index