Recognition of Printed Assamese Characters using Orthogonal Moments

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

Sarat Saharia
Registration No. 021 of 1999

School of Engineering
Department of Computer Science and Engineering
Tezpur University
December, 2009

To the memory of my parents

Abstract

Optical Character Recognition (OCR) is an extensively studied and challenging areas of pattern recognition. Most of the researches on OCR are done for Roman, Chinese, Japanese and Arabic characters. A large number of researches on OCR have also been done on different scripts of Indian languages. The majority of OCR researches on Indian scripts have been done for Devanagari and Bangla scripts. Assamese is one of the Indian languages where very few researches on OCR are reported. The Assamese script is similar to Bangla script but there are few distinct characters in both the languages. Therefore, this research work was taken up to study the representation and recognition aspects of printed Assamese characters.

Statistical moments have been used as features in pattern recognition including character recognition. Most of the studies on character recognition using moments are done for Chinese, Arabic and Roman characters. The moment-based methods for character recognition have a few advantages. Orthogonal moments are becoming more popular due to their better image representation capability than the non-orthogonal moments. However, use of moments for recognition of Indian language scripts is very few. Therefore, we choose to investigate printed Assamese character recognition using orthogonal moments. Three orthogonal moments, namely, Legendre, Tchebichef and Krawtchouk moments are used in this study.

The study includes a comparative analysis of the performance of Legendre, Tchebichef and Krawtchouk moments in representation of printed Assamese characters. The Principal component Analysis (PCA) and the Linear Discriminant Analysis (LDA) are also used for improved representation. The representation aspects of Scale Invariants (SI) of Legendre and Tchebichef moments are studied. Two new moment-based methods for recognition of printed Assamese characters are proposed. The methods extract moments from parts of the character images to reduce ambiguity in recognition by splitting the character images either vertically or horizontally. One of the methods uses LDA and concept of Euler numbers and can be used for both multiple font-sizes and multiple font-types. The other method is computationally less expensive but does not give good recognition accuracy for multiple font-type characters.

Keywords — Optical Character Recognition, Assamese, Moment, Orthogonal Moment, Legendre Moment, Tchebichef Moment, Krawtchouk moment, Principal Component Analysis, Linear Discriminant Analysis, Scale Invariant, Euler Number, Ambiguous Cluster, Unambiguous Cluster

TEZPUR UNIVERSITY

This is to certify that the thesis entitled **Recognition of Printed Assamese Characters using Orthogonal Moments** submitted by Mr. Sarat Saharia to Tezpur University in the Department of Computer Science and Engineering under the School of Engineering in partial fulfillment for the award of the degree of Doctor of Philosophy in Computer Science and Engineering has been examined by us on ..... and found to be satisfactory.

The Committee recommends for the award of the degree of Doctor of Philosophy.

Signature of

Supervisor Co-supervisor External Examiner

Date:

TEZPUR UNIVERSITY

Certificate of the Supervisor

This is to certify that the thesis entitled **Recognition of Printed Assamese Characters using Orthogonal Moments** submitted to Tezpur University in the Department of Computer Science and Engineering under the School of Engineering in partial fulfillment for the award of the degree of Doctor of Philosophy in Computer Science and Engineering is a record of research work carried out by Mr. Sarat Saharia under my supervision and guidance.

All helps received by him from various sources have been duly acknowledged.

No part of this thesis has been reproduced elsewhere for award of any other degree.

Signature of Supervisor

Date: 21-06-2016

Place: Tezpur

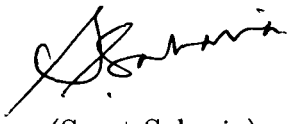
Designation : Professor

School : Engineering

Department : Computer Science and Engineering

Declaration

I, Sarat Saharia, hereby declare that the thesis entitled "*Recognition of Printed Assamese Characters using Orthogonal Moments*" submitted to the Department of Computer Science and Engineering under the School of Engineering, Tezpur University, in partial fulfillment of the requirements for the award of the degree of Doctor of Philosophy in Computer Science and Engineering, is based on bona fide work carried out by me. The results embodied in this thesis have not been submitted in part or in full, to any other university or institute for award of any degree or diploma.



(Sarat Saharia)

Acknowledgements

It is a great pleasure to thank all people who helped and inspired me to make this thesis possible.

I owe my deepest gratitude to my supervisor Prof. D. K. Saikia for his constant guidance throughout the course of this work. He has always been encouraging and made available his great time and effort to make this thesis possible.

I am indebted to Prof. P. K. Bora of IIT Guwahati and would like to acknowledge for his advice, guidance and crucial contributions from the beginning of this work to the end. His perpetual energy and enthusiasm in research was of great help in completing my work.

I also take this opportunity to show my gratitude to Prof. S. N. N. Pandit, Professor (retired) of Osmania University, who introduced me to the pattern recognition area and motivated me to do research in Assamese character recognition when he was visiting professor at this University. I also offer my thanks to Prof. M. Bora of Mathematical Sciences Department for his advice and encouragement during this work.

Prof. D. K. Bhattacharyya, Head of Computer Science and Engineering Department, deserves special thanks for encouragement and valuable advice from time to time. Thanks to Dr. S. M. Hazarika, member of my Doctoral Committee, for his valuable suggestions, advice and help. I also offer my gratitude to Prof. M. Dutta and Prof. S. K. Sinha for their encouragement and advice. Thanks to Dr. U. Sharma, Dr. N. Sarma and all other members of Departmental Research Committee for their valuable suggestions giving proper direction of my work. Thanks to all my colleagues and friends at Department of Computer Science and Engineering for their help and support.

My deepest gratitude goes to my family for their persistent love and support throughout my life. I am indebted to my wife, Jilmil, for her care, love and support. Thanks to my daughter, Dona, for always being a source of inspiration. I am indebted to my parents for all the trouble they took to ensure we get the best in our life. I fondly remember them and offer my sincere tribute to them.

Contents

List of Tables

List of Figures

List of Algorithms

Glossary of Symbols

$(.)^T$	Transpose of a matrix.
$(.)^{+1}$	Pseudo-inverse of a matrix.
$(.)^{-1}$	Inverse of a matrix.
$D(.,.)$	Distance between two clusters.
$D_w(.,.)$	Wighted distance between two clusters.
L_{pq}	Legendre moment of order $(p+q)$.
M	Number of rows of a rectangular image.
N	Number of rows (or columns) of a square image.
N_l	Number of LDA features used for representation.
N_p	Number of Principal Component scores used for representation.
N_s	Number of samples of a dataset.
$P_p(.)$	Legendre polynomial of order p.
Q_{pq}	Krawtchouk moment of order $(p+q)$.
R_r	Radius of the r-th cluster.
T_{pq}	Tchebichef moment of order $(p+q)$.
Δx	Sampling interval in x direction.
Ψ_{pq}	Moment weighting kernel of order $(p+q)$.
$\bar{K}_p(,;.,)$	Weighted Krawtchouk polynomial of order p.
δ_{pq}	Kronecker function.
$\hat{t}_p(.)$	Orthonormal Tchebichef polynomial of order p.
ω^L	Scale invariants of Legendre moments.
ω^T	Scale invariants of Tchebichef moments.
$\parallel \mathbf{x} \parallel$	Euclidean norm of vector $\mathbf{x}$.
σ	Standard deviation of noise (noise level).

$\tilde{t}_p(.)$	Scaled Tchebichef polynomial of order p.
$\widehat{L_{pq}}$	Approximation of L_{pq}.
$\widetilde{L_{pq}}$	Discrete approximation of L_{pq}.
ζ	Domain of a function.
d	Dimension of Euclidean space or number of features.
$d_e(.,.)$	Euclidean distance between two vectors.
k	Number of samples per class of a uniformly distributed dataset.
m_{pq}	Geometric moment of order $(p+q)$.
n	Number of characters(or classes) of an alphabet (or a dataset).
n_r	Number of samples in the r-th class of a non-uniformly distributed dataset.
t	Highest order of moments used for representation.
$t_p(.)$	Tchebichef polynomial of order p.
$t'_{opt,j}$	Optimal order of second stage of the j-th ambiguous cluster of split-character-with-LDA approach.
t_{opt}	Optimal order of first stage of split-character and split-character-with-LDA approach.
t'_{opt}	Optimal order of second stage of split-character approach.
$K_p(,;.,)$	Krawtchouk polynomial of order p.
$\boldsymbol{\mu}_i$	Cluster centroids of first stage of split-character and split-character-with-LDA approach.
$\boldsymbol{\mu}_i{}'$	Cluster centroids of second stage of split-character and split-character-with-LDA approach.
$\mathbb{R}$	The set of real numbers.
$\mathbf{S_b}$	Between-class scatter matrix.
$\mathbf{S_t}$	Total scatter matrix.
$\mathbf{S_w}$	Within-class scatter matrix.
$\mathbf{X_{ri}}$	Feature vector of the i-th sample image of the r-th class of a dataset.
$\bar{\mathbf{X}}_\mathbf{r}$	Centroid vector of the r-th cluster.
$\mathbf{x}$	Feature vector of an input unknown character image.
$\mathbf{G}$	The LDA transformation matrix.
$\mathcal{C}_r$	The r-th cluster.
$\lvert\mathbf{A}\rvert$	Determinant of matrix $\mathbf{A}$.

Chapter 1

Introduction

Optical Character Recognition (OCR) is one of the challenging areas of pattern recognition. Extensive research has been done during last few decades on OCR due to its vast range of applications. OCR research has a comparatively old history in the field of pattern recognition. Invention of retina scanner by Carey in 1870 is considered as the origin of character recognition [88]. The research, during the initial period, was aimed at developing reading machines for the visually handicapped. The first concept of the idea of OCR, as reported in the literature, appeared in two patents obtained by Tausheck [147] in Germany in 1929 and by Handel [55] in the U.S. in 1933. However, the development of a machine that could read characters and numerals was not realized until the arrival of age of computers in the 1950's. The commercial OCRs of first generation appeared in the beginning of the 1960s using logical template matching methods. From these initial developments, the field of OCR research has gone through tremendous developments in terms of different method being employed and type of characters the OCRs can read. The OCRs of present generation are characterized by complex documents containing text, graphics, mathematical symbols, tables, handwritten characters, low-quality noisy documents etc. However, the present state of OCR research is not same for all scripts or languages. Though, the research works in OCR for some scripts are at very advanced stage, they are in developing stage for many other scripts.

1.1 Motivation

Most of the research works reported in the area of OCR are for Roman, Chinese, Japanese and Arabic scripts. The amount of research works on recognition of characters of Indian scripts is not sufficient considering the variety of Indian scripts. Assamese is one of the Indian languages where very few OCR research works are reported in the literature. Though Assamese script is similar to Bangla script, there are a few distinct characters in both languages. Considerable research has been done on Bangla script but exclusive works on Assamese character recognition are very few. Though the methods proposed for Bangla scripts may be extended for Assamese, the author feels that it is worthwhile to carry out study using other techniques on Recognition of Assamese characters which may also be extended to any other scripts including Bangla.

Statistical moments have been used as features in pattern recognition including character recognition, image classification, target identification, scene analysis etc. [113], [43]. Moment-based character recognition methods for one script can easily be adapted to another script because the methods are developed without considering the specific features of a script. Moments are also computationally less expensive than other filter bank based methods where the features are extracted by applying the filters in different directions and scales. Recently, orthogonal moment-based features are getting more attention due to less redundancy and better image representation capabilities in comparison to non-orthogonal moments [86], [97], [157], [159]. Most of the studies on moment based character recognition reported in the literature are on Chinese characters. To the best of the author's knowledge, very few studies on Indian scripts recognition using orthogonal moments are reported in the literature. It is, therefore, worthwhile to investigate the suitability of orthogonal moments as features for recognition of Indian scripts. Among the orthogonal moments defined in continuous domain, Zernike and Legendre moments are more popular. In this study we consider only Legendre moments among the continuous moments because Zernike moments are computationally more complex [99]. Since the discrete orthogonal moments are known to be better than the continuous orthogonal moments, we use two most popular discrete orthogonal moments

viz., Tchebichef and Krawtchouk. Similar study may also be done using other moments.

1.2 Objectives of the Work

In pattern recognition problems appropriate representation of the patterns itself is very important to solve the problem of recognition. Therefore our first effort is to study the problem of appropriate representation. This is followed by development of methods for recognition. The objectives of this thesis are as follows:

1. To study the appropriateness of representing printed Assamese characters with orthogonal moments like Legendre, Tchebichef and Krawtchouk.
2. To study the performance of Scale Invariants of Legendre and Tchebichef moments for representation of:
   (a) Printed Assamese characters in presence of noise, and
   (b) Printed Assamese characters of multiple font-sizes.
3. To develop new methods based on orthogonal moments for recognition of printed Assamese characters of multiple font-sizes and multiple font-types.

1.3 Organization of the Thesis

Apart from this introductory chapter, the thesis includes five more chapters. These chapters are briefly outlined below.

1.3.1 Chapter 2: Optical Character Recognition

This chapter briefly reviews the development of Optical Character recognition. It starts with some of the major developments in OCR during the initial days [55], [93], [88], [147]. The generations of commercial OCR systems depending on versatility, robustness and efficiency are also discussed. This chapter also briefly reviews the development of OCR for Indian scripts. Amongst the existing

works on Indian scripts, majority of works are done on Devanagari and Bangla scripts. Some other works, reported in the literature, on recognition of other Indian languages include Tamil, Telugu, Oriya, Kannada, Gujarati, Gurmukhi, etc. Most of the works on Indian script OCR use spatial domain features but use of transform domain features, moment-based features, ANN, SVM etc. are very few.

Finally, the chapter discusses the use of statistical moments for character recognition. Most of the studies on moment based character recognition reported in the literature are on Chinese characters but such studies are very few for Indian scripts.

1.3.2 Chapter 3: Representation of Printed Assamese Characters using Orthogonal Moments

This chapter presents the study on representation of printed Assamese character using orthogonal moments, namely, the Legendre, the Tchebichef and the Krawtchouk. A comparative analysis of their performance is also carried out. The study presented in this chapter also includes use of Principal Component Analysis (PCA) and Linear Discriminant Analysis for better representation.

1.3.3 Chapter 4: Representation of Printed Assamese Characters using Scale Invariants of Orthogonal Moments

Scale Invariants (SI) of Legendre and Tchebichef moments has been reported in [28] and [160] respectively. In this chapter, a study on representation of printed Assamese characters using the Scale Invariants (SI) of Legendre and Tchebichef moments is presented. The representation aspects of the scale invariants are investigated for printed Assamese characters in presence of noise and for characters of multiple font-sizes.

1.3.4 Chapter 5: Recognition of Printed Assamese Characters by Splitting of Characters using Orthogonal Moments

In this chapter, two new moment-based methods for recognition of printed Assamese characters are proposed. Both the methods split character images either vertically or horizontally, in some situations where ordinary moment methods fail to distinguish different characters, and extract moment features from the appropriate half as distinguishing features. The first of these two methods is computationally less expensive but not suitable for multiple font-type characters. The second method, which is computationally more expensive, uses Linear Discriminant Analysis and concept of Euler numbers, in some cases and is suitable for characters of multiple font-sizes and multiple font-types.

1.3.5 Conclusions and Future Work

The last chapter summarizes the contributions of this work and also gives directions for future work.

Chapter 2

Optical Character Recognition

2.1 Introduction

Optical Character Recognition (OCR) is one of the challenging areas of pattern recognition. It gained popularity among the research community due to its vast application potentials. Extensive research has been done on OCR evidenced by a large number of research articles published in the literature during the last few decades. Most of the research works reported in this area are for Roman, Chinese, Japanese and Arabic scripts. The number of research works on recognition of characters of Indian scripts is not sufficient keeping in view the variety of Indian scripts. There are 12 major scripts for 16 major Indian languages. There are many commercial OCRs available nowadays and most of them work for Roman, Chinese, Japanese and Arabic characters. This chapter presents an overview of OCR research and development. OCRs on Indian language scripts are given more emphasis in this chapter and special attention is given to the statistical moment-based character recognition methods.

2.2 Optical Character Recognition

The process of automatic recognition of characters in optically scanned and digitized document text is called Optical Character Recognition (OCR). It is a process of translating human-readable characters to machine readable codes.

It can improve the man-machine interface and can contribute greatly to the automation process in many applications.

OCR has many practical applications including the following:

1. Banks and Post Offices: automatic reading and sorting of mail, bank cheques and such other documents
2. Library: automatic text entry into the computer for cataloging, ledgering etc.
3. Desktop publishing: automatic reading from scanned documents for editing and publishing
4. Document data compression: converting scanned document images to ASCII and other computer readable format
5. Reading aid for blind: OCR can be combined with a text-to-speech converter which can read out text from a scanned document
6. Language processing: automatic text entry from manuscripts for language processing

2.3 Development of OCR

The invention of retina scanner by C. R. Carey of Boston Massachusetts in the year 1870, which was a image transmission system using a mosaic of photocells, is considered as the origin of character recognition [88]. The second significant event in the history of OCR is the invention of sequential scanner by Nipkow in 1890 which was a major breakthrough for modern television and reading machines. During the initial days the efforts were primarily made towards the development of character recognition devices as an aid to the visually handicapped and successful attempts were made by the Russian scientist Tyurin in 1900. Two patents on OCR were obtained: one in 1929 by Tausheck in Germany [147] and the other in 1933 by Handel in the U.S. [55]. These are the first concepts of the

idea of OCR as reported in the literature [93]. The principle of Tausheck's patent is template/mask matching.

The commercial OCR systems are divided into four generations depending on versatility, robustness and efficiency [107], [93], [38]. The first generation systems are characterized by the constrained letter shapes which the OCRs read. Such machines appeared in the beginning of the 1960s. Two most typical systems were NCR 420 [49] and Farrington 3010 [64]. IBM developed the first widely commercialized OCR of this generation, the IBM 1418 [53], which was designed to read a special IBM font, 407 [93] . The recognition method was logical template matching where the positional relationship was fully utilized. Some other OCR systems of IBM that follow are IBM 1428, IBM 1285 and IBM 1975. By the end of the 1960's some mainframe companies of Japan announced their OCRs of the first generation. Some of them are Facom 6300A from Fujitsu, H-852 from Hitachi and N240D-1 of NEC. The first two systems used stroke analysis method and the third one used logical template matching [93] .

The second generation OCR systems appeared in the middle of 1960s to early 1970s and they are characterized by the recognition capabilities of a set of hand-printed characters in addition to regular machine printed characters [93], [107]. At the early stages of this generation, the scope was restricted to recognition of numerals only. IBM 1287 was the first and famous OCR system of this generation which was exhibited at the 1965 New York world fair. The system was developed as a hybrid one combining analog and digital technology. Another significant development during this period is the first automatic letter-sorting machine for postal code numbers of Toshiba. The methods were based on the structural analysis approach [48], [93], [107]. Some other OCR systems of this generation was RETINA of Recognition Equipment Inc.[135], H-8959 of Hitachi [129].

The third generation OCR systems were targeted to recognize characters from poor-print-quality documents, and hand-printed characters for a large category character set. Commercial OCR systems, which partially met these targets, appeared roughly during the period from 1975 to 1985. Two common objectives for these systems were high performance and low cost [93], [94], [102].

The fourth generation is characterized by the OCRs of complex documents

containing text, graphics, table and mathematical symbols, unconstrained handwritten characters, color document, low-quality noisy documents like photocopy and fax, etc [107]. Due to the upgraded performances of UNIX workstations and personal computers, a major part of the recognition was implemented by softwares on general purpose computers and commercial OCR software packages running on personal computers also appeared in the market [45]. Advanced layout analysis techniques enabled recognition of a large variety of complex documents. There are many works reported in the literature on unconstrained handwritten character but the recognition accuracy hardly exceeds 85% [107]. On the otherhand very few studies are reported on color documents and research on this problem is continuing [107]. Another area of research in this period is recognition of characters on noisy documents [114], [143]. The commercial products of this period include postal address readers that can sort automatically hand-printed addresses [103] and a reading aid for the blind marketed by XeroxKurzweil which is an integrated OCR with speech output for English language [74].

At present, more sophisticated optical readers which can process documents that has been typewritten, typeset, or printed by dot-matrix, line and laser printers are available for Roman, Chinese, Japanese and Arabic text. These readers can recognize characters with different fonts and sizes, with different formats including intermixed text and graphics [93], [6], [40], [41], [101], [51], [123], [144]. With the introduction of narrow range scanners, an optical reader can recognize multiple columns or sections of a page or mailing lists. Some of them are also equipped with software for spell checking, and for flagging suspicious characters or words [114], [150]. More recently, several research works are reported in the literature in areas of low-quality, degraded character recognition [132], [11], [87], [59], recognition of historical documents [75], [120], [2], [70], [152], vehicle number plate recognition [21], [7], [63], recognition of multilingual documents [23], [62], [31], [92], [142], [15] etc.

The the characteristics and period of these OCR generations are shown in Table 2.1.

Table 2.1: Generations of OCR

Generation	Period	Type of Document/Character OCRs Read
First	Beginning of the 1960s	Constrained letter shapes
Second	Mid 1960s to early 1970s	Hand-printed and machine-printed characters
Third	1975 - 1985	Poor quality documents, large set of hand-printed characters
Fourth	1985 -	Complex documents containing text, graphics, tables, mathematical symbols, unconstrained handwritten character set etc.

2.4 Methodologies used for OCR

The Optical Character Recognition techniques are broadly classified as template-based and feature-based approaches [107], [93]. The template-based approach, which is also called template-matching approach, was used in OCR systems during the early stage of development. In this approach, an input character image is superimposed on a template and then the degree of correlation between them is measured and this measure is used for classification. Applications of template-matching approach are also found in modern OCR systems in combination with feature-based methods [25].

Feature-based methods extract important characteristics or features from the characters and use them to develop classification models. There are two types of feature-based approaches: spatial domain approach and transform domain approach. Spatial domain approaches use features which are directly extracted from the spatial representation of the pixels. Examples of spatial domain features are stroke-based feature, topological features, features obtained from the concept of water reservoirs [26] etc. The techniques under the transform domain approach, first transform a character image into another space like Fourier, Cosine, Wavelet and then useful features are obtained from the transformed images. These

features are then used in a sophisticated classification model. Other techniques used for OCR include graph-theoretic [65] and formal grammar-based approaches [42]. A large number of works are also done on moment-based OCR [18], [39], [156].

Among the modern techniques, Artificial Neural Network (ANN), Hidden Markov Model (HMM), Support Vector Machines (SVM), Fuzzy rules, tolerant rough set, Evolutionary algorithms are also used for character recognition. In these techniques, the systems do not explicitly derive any feature from the characters. Such a system is trained by raw or normalized patterns and the system adjusts itself to have best possible classification of these patterns. The trained system is used to classify unknown patterns of character images.

2.5 Works on Indian Script Optical Character Recognition

There are 22 languages recognized by the Constitution of India, spoken in different parts of the country, namely Assamese, Bengali, Bodo, Dogri, Gujarati, Hindi, Kannada, Kashmiri, Konkani, Maithili, Malayalam, Meitei, Marathi, Nepali, Oriya, Punjabi, Sanskrit, Santhali, Sindhi, Tamil, Telugu and Urdu [161], [162]. These languages are written using twelve different scripts. Most of these scripts originated from ancient Brahmi through various transformations [36]. Considering these large number of languages and scripts, the number of studies on Indian language script character recognition is not sufficient. Amongst the existing works on Indian scripts, majority of works are done on Devanagari and Bangla scripts. Some other works, reported in the literature, on recognition of other Indian languages include Tamil, Telugu, Oriya, Kannada, Punjabi, Gujarati, etc.

A large number of works are done for OCR in Devanagari script. Among the early works, some efforts were made for Devanagari character recognition by Sinha [137], [139]. In his doctoral thesis [137], he discussed a syntactic pattern analysis system and its application to Devanagari script recognition. A system for recognition of handwritten and machine printed Devanagari characters was

presented in [139]. Another important work of Sinha is found in [138], where he demonstrated how the spatial relationship among the constituent symbol of Devanagari scripts plays an important role in interpretation of Devanagari words. Some other initial studies on Devanagari script recognition was also done in [133], [134]. The work in [137] discusses a Devanagari hand-printed numeral recognition system on the basis of presence of some basic primitives, namely, horizontal and vertical line segment, left and right slant, curves etc. and their positions and interconnections. A similar technique was presented in [134] for recognition of constrained hand-printed Devanagari characters. The above systems deal with character recognition in isolation. Two first works on complete OCR systems for printed Devanagari characters are due to Palit and Choudhury [112] and Pal and Choudhury [105]. The method proposed in [105] is reported to give 96% accuracy. Recently, a system for recognition of hand-written Devanagari numeral recognition is proposed in [12]. Here, two types of features are used to represent the numerals and a multilayer perceptron is used for categorization of the numerals.

Bangla is another Indian script where lots of research works are reported. Some early works on recognition of Bangla characters are done in [22], [113], [141]. A nearest neighbour classifier for Bangla character recognition is presented in [122], which uses features by string connectivity criterion. During the same time, a generalized formal approach for generation and analysis of Bangla and Devanagari characters is presented in [36]. The first work on a complete Bangla OCR is done by Choudhury and Pal [25]. The system uses both feature and template matching for recognition. A feature-based tree classifier is used for recognition of simple characters and the compound/conjunct characters are recognized using run-based template matching which is preceded by feature-based grouping. The system is reported to have almost 96% recognition accuracy. Skew angle detection and correction is an important preprocessing step in any OCR system. A novel technique for skew detection and correction is proposed in [24]. An algorithm to estimate the skew angle of individual text lines in a document where different text lines may have different skew is proposed in [110]. Error detection and correction is an important aspect of an OCR system. An error detection and correction technique based on morphological parsing of

recognized word is presented in [109]. Works on handwritten Bangla character recognition are also reported in the literature. Some of these works are: an isolated handwritten numeral recognition [113], a multistage classification scheme for handwritten Bangla character recognition [117], a robust scheme for off-line recognition of Bangla isolated handwritten numerals [106], a neural network-based system for recognition of isolated handwritten alphanumeric characters [37], etc. Some of the recent works are: a water reservoir based scheme for the segmentation of unconstrained handwritten texts into line, words and characters [108], a neural network approach for recognition of Bangla handwritten numerals [16], on-line recognition of Bangla handwritten characters [47] etc.

Tamil is another language where some amount of work on character recognition is reported. Some of the early works on recognition of machine-printed and hand-printed Tamil character recognition is found in [140], [27], [20]. A work on on-line Tamil character recognition is reported in [146].

Among the early works on Telugu character recognition, a two-stage printed Telugu character recognition system is reported in [118]. The system uses a knowledge-base about the primitive shapes present in the characters and classification is done by a decision tree. Another printed Telugu character recognition system by neural network approach is proposed in [145]. A compositional approach for printed Telugu character recognition using connected components and fringe distance based template matching is proposed in [104].

Some research works are reported for recognition of Oriya characters. One system to recognize Oriya alphabet using Kohonen neural network is proposed in [91]. The system was tested only on five Oriya characters and hence the reliability of the system was not established. A complete OCR system for printed Oriya characters is presented in [26]. The system uses a combination of stroke and run-number-based features along with features obtained from the concept of water overflow from a reservoir. The feature detection methods are simple and robust and it is reported to have about 96% accuracy.

A complete OCR system for printed Gurmukhi script is reported in [77]. The system uses a multi-stage classification scheme combined with binary tree and nearest neighbour classifier. The system is reported to give about 97.34%

accuracy. Some other studies on recognition of Gurmukhi characters are found in [81], [80], [79], [82], [78]. An post-processor for Gurmukhi script OCR is also proposed in [76].

To the best of the author's knowledge, the first work on recognition of Gujarati characters is reported in [8], which describes classification of a subset of printed Gujarati characters. Another work on zone identification of printed Gujarati text is reported in [32].

Another Indian language, where a few research works on character recognition are available, is Kannada. Two recent works on Kannada character recognition are: a font and size independent OCR system reported in [9], and an on-line system for handwritten Kannada character recognition using Wavelet features and neural classifier reported in [119].

Among few research works in Assamese OCR, a method for generating a feature set using neural feature extractor for use in neural network based Assamese character and numeral recognition is reported in [131]. Another work on Assamese character and Anglo-Assamese numeral recognition using hybrid feature set is reported in [130]. A method for handwritten Assamese character segmentation using Artificial Neural Network is reported in [17]. Representation of printed Assamese characters using Legendre and Tchebichef moments are studied in [124], [125].

2.6 Optical Character Recognition using Moments

The first work on character recognition by moments was done by M.K.Hu [58]. He demonstrated the utility of invariant features based on geometric moments for pattern recognition. Since then, moment-based features are used in many diverse areas like pattern recognition including character recognition, image classification, target identification, scene analysis [116], [43]. Application of moments in recognition of aircraft silhouette is described in [14] and [34]. The moments and moment invariants are also used in template matching and

registration of satellite images [155], [52], [43], texture classification [154], and character recognition [14], [13], [44], [66], [10], [1], [67], [69], [68], [83], [40]. Most of the studies on moment based character recognition reported in the literature are on Chinese, Arabic and Roman characters [83], [84], [85], [86], [6], [40], [39], [3], [5], [14], [58], [68], [67], [69], [1], [35], [10]. To the best of the author's knowledge, very few studies on Indian scripts are reported in the literature [72], [125], [8], [73]. Representation of printed characters using orthonormal Tchebichef moments and Legendre moments are studied in [128], [124], [125]. Some other studies on orthogonal Legendre moments as features are reported in [85], [10], [28]. Orthogonal Zernike moment is used in recognition of handwritten Devanagari characters in [72]. Recently some discrete orthogonal moments like Tchebichef[97], Krawtchouk[157], Racah[159] based on discrete orthogonal polynomials are also used as feature descriptors and are reported as better alternatives for continuous orthogonal moments.

2.7 Conclusion

In this chapter, an overview of research and development of OCR, which is one of the most challenging and extensively studied areas of pattern recognition, is presented. At first, some of the major developments in OCR during the initial days are presented. The generations of commercial OCR systems depending on versatility, robustness and efficiency are also discussed. Most of the commercial OCR systems are meant for Roman, Chinese, Japanese and Arabic characters. The chapter also briefly reviews the developments of OCR for Indian scripts. Considering the large number of languages and scripts in India, the number of research works in Indian language OCR is not sufficient. The majority of research works are done on Bangla and Devanagari scripts. Finally, the chapter discusses the use of statistical moments for character recognition. Most of the studies on moment-based character recognition reported in the literature are on Roman, Chinese and Arabic characters but such studies are very few for Indian scripts.

Chapter 3

Representation of Printed Assamese Characters using Orthogonal Moments

3.1 Introduction

Moment functions are widely used in image analysis as feature descriptors. Compared to geometric moments, orthogonal moments have become more popular in image analysis for their better representation capabilities. In comparison to continuous orthogonal moments discrete orthogonal moments provide a more accurate description of the image features. This chapter presents a study on representation of printed Assamese characters using orthogonal moments, namely, Legendre, Tchebichef and Krawtchouk and also gives a comparative analysis of their performance. The chapter also discusses the use of Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) for improved representation.

3.2 Moment Functions

A digital image of a character or any other object can be defined as a two-dimensional density distribution $f(x, y)$, where the function values are the

intensities at pixel locations (x, y). A general definition of moment functions Φ_{pq} of order $(p+q)$, of the function $f(x, y)$ is given by

$$\Phi_{pq} = \iint_{\zeta} \Psi_{pq}(x, y) f(x, y)\, dx\, dy, \quad p, q = 0, 1, 2, 3, \ldots \quad (3.1)$$

where ζ denotes the domain of the function $f(x, y)$ and $\Psi_{pq}(x, y)$ is a continuous function of (x, y) in ζ, known as the *moment weighting kernel* or the *basis set* [100]. The indices p, q usually denote the degrees of the coordinates x, y respectively, as defined inside the function Ψ. The intensity function $f(x, y)$ of an image is bounded and has compact support in ζ and therefore, the integral given by Equation 3.1 is finite.

3.2.1 Problems Associated with Implementation of Moment Functions

The implementation of moment functions involve the following major problems [98]:

1. *Numerical approximation of continuous integrals:* The integrals in the definition of two-dimensional moment function of Equation 3.1 are usually approximated by discrete summations. This leads to numerical errors in the computed moments and also severely affects the analytical properties such as invariance, orthogonality etc., which they are intended to satisfy.

2. *Large variations in the dynamic range of values:* Because of the powers p and q of the kernel Ψ_{pq} of Equation 3.1, the computed moments have large variations in the dynamic range of values for different orders. Therefore, it is necessary to include additional measures, like scale normalization, in applications involving these moments to maintain equal weight for all components in a set of feature vectors.

3. *Coordinate space transformation:* Generally, the domains of the moment kernels are completely different from the image coordinate space. Therefore, applications of such kernels in Equation 3.1 require an appropriate

transformation of the image coordinate space which increases the computational complexity [100].

The continuous orthogonal basis functions, such as Legendre polynomial, Zernike polynomial etc., do not have the above problem of large dynamic range variation but generally have the other two problems. On the other hand, discrete orthogonal polynomials as basis set overcome all the above problems associated with implementation of moment functions. Some examples of discrete orthogonal basis functions are Tchebichef, Krawtchouk, Racah etc.

3.2.2 Geometric moments

The geometric moments are the simplest of the moment functions. The $(p+q)th$ order two-dimensional geometric moments, denoted by m_{pq}, are defined with the basis set $\{x^p y^q\}$, and can be expressed as

$$m_{pq} = \iint_{\zeta} x^p y^q f(x,y)\, dx\, dy \tag{3.2}$$

Geometric moments are widely used as object descriptors. The following are two important theorems about geometric moments [58], [100].

Uniqueness Theorem: Assuming that the intensity function $f(x,y)$ is piece-wise continuous and bounded in the region ζ, the moment sequence $\{m_{pq}\}$ is uniquely determined by the intensity function $f(x,y)$, and conversely.

Existence Theorem: Assuming that the intensity function $f(x,y)$ is piece-wise continuous and bounded in the region ζ, the moments m_{pq} of all orders exists and are finite.

3.2.3 Legendre Moments

Legendre moments were introduced as image processing tools by Teague [148]. The two-dimensional Legendre moments of order $(p+q)$ of an image intensity function $f(x,y)$, are defined as [100]

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{+1} \int_{-1}^{+1} P_p(x)\, P_q(y) f(x,y)\, dx\, dy, \tag{3.3}$$

where $P_p(x)$ is the Legendre polynomial of order p and is given by

$$P_p(x) = \frac{1}{2^p p!}\frac{d^p}{dx^p}(x^2-1)^p, \quad x \in [-1,1], \tag{3.4}$$

which hold the following orthogonal property:

$$\int_{-1}^{+1} P_p(x)P_q(x) = \frac{2}{2p+1}\delta_{pq} \tag{3.5}$$

where δ_{pq} is the Kronecker function, i.e., $\delta_{pq} = 1$ if $p = q$ and 0 otherwise.

3.2.3.1 Computation of Legendre Moments

The Legendre polynomials are defined in the range $[-1,1]$. Therefore, computation of these moments requires the image coordinate space to be transformed to the range $[-1,1]$. At the same time the continuous integrals are to be approximated by discrete summations.

For a digital image of size N^2, an approximation of L_{pq} is given by

$$\widetilde{L_{pq}} = \frac{(2p+1)(2q+1)}{(N-1)^2}\sum_{i=1}^{N}\sum_{j=1}^{N} P_p(x_i)P_q(y_j)f(i,j), \tag{3.6}$$

where x_i, y_j denote the normalized pixel coordinates in the range $[-1,1]$, given by

$$\begin{aligned} x_i = (2i/N) - 1; \quad y_j = (2j/N) - 1 \\ i,j = 1,2,\ldots,N \end{aligned} \tag{3.7}$$

The computation of Legendre moment can be facilitated by using the following recursive relation in Legendre polynomials [100]:

$$P_p(x) = \frac{(2p-1)xP_{p-1}(x) - (p-1)P_{p-2}(x)}{p} \tag{3.8}$$

where $P_0(x) = 1$; $P_1(x) = x$; $|x| \leq 1$ and $p > 1$.

An efficient algorithm for fast computation of Legendre moments is given in [100].

The implementation of Legendre moments using Equation 3.6 involves some approximation errors due to the problems mentioned in 1 and 3 of Subsection 3.2.1. These approximation errors increase with the order of moments [86]. Therefore higher order moments fail to represent detail features of the images.

A better approximation of L_{pq} is proposed in [86] and is given by the following equation:

$$\widehat{L_{pq}} = \frac{(2p+1)(2q+1)}{(N-1)^2} \sum_{i=1}^{N} \sum_{j=1}^{N} h_{pq}(x_i, y_j)\, f(i,j), \tag{3.9}$$

where

$$h_{pq}(x_i, y_j) = \int_{x_i - \frac{\Delta x}{2}}^{x_i + \frac{\Delta x}{2}} \int_{y_j - \frac{\Delta y}{2}}^{y_j + \frac{\Delta y}{2}} P_p(x) P_q(y)\, dx\, dy, \tag{3.10}$$

$\Delta x = x_i - x_{i-1} = 2/N$ and $\Delta y = y_j - y_{j-1} = 2/N$ are the sampling intervals in x and y directions [86].

The Equation 3.10 gives the integration of $P_p(x)P_q(y)$ over the (i,j)th pixel. It has been shown that the moments computed using Equation 3.9 contain less approximation errors than using Equation 3.6 [126]. Use of Equations 3.9 and 3.10 requires working out the Legendre polynomials which is not feasible for any arbitrary order. We propose to use the recursive relation given in Equation 3.8 and the following integral formula for Legendre polynomials [100] in Equations 3.9 and 3.10:

$$\int P_p(x)dx = \frac{xP_p(x) - P_{p-1}(x)}{p+1} \tag{3.11}$$

For images of size $M \times N$ the Equation 3.9 is modified to

$$\widehat{L_{pq}} = \frac{(2p+1)(2q+1)}{(M-1)(N-1)} \sum_{i=1}^{M} \sum_{j=1}^{N} h_{pq}(x_i, y_j)\, f(i,j), \tag{3.12}$$

and the sampling intervals in Equation 3.10 become $\Delta x = x_i - x_{i-1} = 2/M$ and $\Delta y = y_j - y_{j-1} = 2/N$.

3.2.4 Tchebichef Moments

Tchebichef moments are based on discrete orthogonal Tchebichef polynomials and are introduced in [98] as tools for analysis of two-dimensional images. The $(p+q)$ order Tchebichef moments of an image intensity function $f(x,y), 0 \leq \{x,y\} \leq N-1$ are defined as [98]

$$T_{pq} = \frac{1}{\rho(p,N)\rho(q,N)} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \tilde{t}_p(x)\tilde{t}_q(y)\, f(x,y), \qquad p,q = 0,1,2,\ldots N-1 \tag{3.13}$$

where $\rho(p,N)$ is the squared norm and $\tilde{t}_p(x)$ is the scaled Tchebichef polynomial and is defined as

$$\tilde{t}_p(x) = \frac{t_p(x)}{\beta(p,N)} \tag{3.14}$$

and $t_p(x)$ is the discrete Tchebichef polynomial of order p and is given by

$$t_p(x) = p! \sum_{k=0}^{p} (-1)^{n-k} \binom{N-1-k}{p-k} \binom{p+k}{p} \binom{x}{k}, \qquad n,x = 1,2,3,\ldots,N-1 \tag{3.15}$$

and $\beta(p,N)$ is the scale factor which is independent of x. The Tchebichef polynomials satisfy the following orthogonality condition

$$\sum_{x=0}^{N-1} t_p(x)t_q(x) = \rho(p,N)\delta_{pq}, \qquad 0 \leq p,q \leq N-1 \tag{3.16}$$

where δ_{pq} is the Kronecker function and the squared norm $\rho(p,N)$ is given by

$$\begin{aligned} \rho(p,N) &= \frac{N(N^2-1)(N^2-2)\ldots(N^2-p^2)}{2p+1}, \\ &= (2p)!\binom{N+p}{2p+1}, \quad p = 0,1,2,\ldots,N-1 \end{aligned} \tag{3.17}$$

The scale factor $\beta(p,N)$ in Equation 3.14 is typically a function of N and the simplest choice is

$$\beta(p,N) = N^p \tag{3.18}$$

With this choice for $\beta(p, N)$, the squared norm $\tilde{\rho}(p, N)$ is modified to

$$\tilde{\rho}(p, N) = \frac{N(1 - \frac{1^2}{N^2})(1 - \frac{2^2}{N^2}) \ldots (1 - \frac{p^2}{N^2})}{2p + 1} \tag{3.19}$$

The problem associated with Equation 3.13 is that, the value of the squared norm tends to zero as p increases. As a result, Equation 3.13 gives very large values for the moments when either p or q is large. The problem can be resolved by the orthonormal version of the moments [95], [96]. The orthonormal version of Tchebichef polynomials can be obtained by modifying the scale factor $\beta(p, N)$ as

$$\begin{aligned} \beta(p, N) &= \sqrt{\rho(p, N)} \\ &= \sqrt{\frac{N(N^2 - 1)(N^2 - 2^2) \ldots (N^2 - p^2)}{2p + 1}} \end{aligned} \tag{3.20}$$

With the above scale factor, the orthonormal Tchebichef polynomials, denoted by $\{\hat{t}_i\}$, are given by the following recurrence relation:

$$\begin{aligned} \hat{t}_p(x) &= \alpha_1 x \hat{t}_{p-1}(x) + \alpha_2 \hat{t}_{p-1}(x) + \alpha_3 \hat{t}_{p-2}(x), \\ p &= 2, 3, \ldots, N - 1; x = 0, 1, 2, \ldots, N - 1 \end{aligned} \tag{3.21}$$

where

$$\begin{aligned} \alpha_1 &= \frac{2}{p}\sqrt{\frac{4p^2 - 1}{N^2 - p^2}} \\ \alpha_2 &= \frac{(1 - N)}{p}\sqrt{\frac{4p^2 - 1}{N^2 - p^2}} \\ \alpha_3 &= \frac{(p - 1)}{p}\sqrt{\frac{2p + 1}{2p - 3}}\sqrt{\frac{N^2 - (p - 1)^2}{N^2 - p^2}} \end{aligned} \tag{3.22}$$

The initial conditions of the above recurrence relation are

$$\begin{aligned} \hat{t}_0(x) &= N^{-\frac{1}{2}} \\ \hat{t}_1(x) &= \frac{\sqrt{3}(2x + 1 - N)}{\sqrt{N(N^2 - 1)}} \end{aligned} \tag{3.23}$$

The discrete orthonormal polynomials defined as above satisfy the following condition for all p:

$$\hat{\rho}(p, N) = \sum_{i=0}^{N-1} \{\hat{t}_p(i)\}^2 = 1.0 \tag{3.24}$$

where $\hat{\rho}(p, N)$ is the squared norm of $\{\hat{t}_p(i)\}$.

With these normalized Tchebichef polynomials the Equation 3.13 is now reduced to

$$T_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \hat{t}_p(x)\hat{t}_q(y)f(x, y), \quad p, q = 0, 1, 2, \ldots N-1 \tag{3.25}$$

For images of size $M \times N$, the Equation 3.25 can be modified as follows:

$$T_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \hat{t}_p(x)\hat{t}_q(y)f(x, y), \quad p = 0, 1, 2, \ldots M-1; q = 0, 1, 2, \ldots, N-1 \tag{3.26}$$

In order to reduce propagation of any numerical errors through the recurrence relation, the polynomials given in Equation 3.21 can be renormalized as follows:

$$\hat{t}_p(x) \leftarrow \frac{\hat{t}_p(x)}{\sqrt{\sum_{x=0}^{N-1} (\hat{t}_p(x))^2}} \tag{3.27}$$

It has been shown that the reconstruction accuracy improves significantly by renormalization of the orthonormal Tchebichef polynomials [95].

We use the renormalized version of the orthonormal Tchebichef polynomial in our study.

3.2.4.1 Computation of Tchebichef moments

The basis functions of Tchebichef moments are orthogonal in the discrete domain of the image coordinate space. Therefore, implementations of these moments do not require any discrete approximation. It also eliminates the need for coordinate space normalization. The Equations 3.25 and 3.26 can be used directly to

compute the Tchebichef moments. Due to the use of appropriate scale factors, the computed moments do not exhibit large variation in the dynamic range of values. It also avoids any kind of numerical instabilities for large image sizes. Therefore, implementation of Tchebichef moments overcomes all the problems mentioned in Subsection 3.2.1 and they are proved to be superior to the conventional orthogonal moments like Legendre and Zernike moments.

3.2.5 Krawtchouk Moments

The Krawtchouk moments are based on discrete classical Krawtchouk polynomials and are introduced in [157] as tools for two-dimensional image analysis. The two-dimensional Krawtchouk moments of order $(p+q)$ for a an image intensity function $f(x,y), 0 \leq \{x,y\} \leq N-1$ are defined as [157]

$$Q_{pq} = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \bar{K}_p(x; t_1, N-1) \bar{K}_q(y; t_2, N-1) f(x,y) \tag{3.28}$$

where $\bar{K}_p(x; t, N-1)$ is the pth order weighted Krawtchouk polynomial and is defined as

$$\bar{K}_p(x; t, N-1) = K_p(x; t, N-1) \sqrt{\frac{w(x; t, N-1)}{\rho(p; t, N-1)}} \tag{3.29}$$

where $K_p(x; t, N-1)$ is the pth order discrete Krawtchouk polynomial defined as

$$\begin{gathered} K_p(x; t, N-1) = \sum_{k=0}^{p} a_{k,p,t} x^k = {}_2F_1(-p, -x; -(N-1); \frac{1}{t}) \\ x, p = 0, 1, 2, \ldots, N-1; N-1 > 0; t \in (0,1), \end{gathered} \tag{3.30}$$

${}_2F_1$ is the hypergeometric function and is defined as

$${}_2F_1(a, b; c; z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!} \tag{3.31}$$

and $(a)_k$ is the Pochhammer symbol given by

$$(a)_k = a(a+1)\dots(a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)} \tag{3.32}$$

The set of Krawtchouk polynomials defined by Equation 3.30 forms a complete set of discrete basis functions with weight function

$$w(x;t,N-1) = \binom{N-1}{x} t^x (1-t)^{N-1-x} \tag{3.33}$$

and satisfies the orthogonality condition

$$\sum_{x=0}^{N-1} w(x;t,N-1)K_p(x;t,N-1)K_q(x;t,N-1) = \rho(p;t,N-1)\delta_{pq} \qquad p,q = 1,2,\dots,N-1 \tag{3.34}$$

where

$$\rho(p,t,N-1) = (-1)^p \left(\frac{1-t}{t}\right)^p \frac{p!}{(-(N-1))_p} \tag{3.35}$$

and δ_{pq} is the Kronecker function.

For images of size $M \times N$, the Equation 3.28 becomes

$$Q_{pq} = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} \bar{K}_p(x;t_1,M-1)\bar{K}_q(y;t_2,N-1)f(x,y) \tag{3.36}$$

Krawtchouk moments have the additional property of being able to extract local features from any region-of-interest by varying the parameter t of the binomial distribution associated with the Krawtchouk polynomial [157].

3.2.5.1 Computation of Krawtchouk moments

In order to avoid overflowing for mathematical functions like hypergeometric functions and gamma functions, recurrence relation can be used in computation of Krawtchouk polynomials. The three-term recursive relation for the weighted Krawtchouk polynomials is given by [157], [153]

$$\bar{K}_{p+1}(x;t;N-1) = \frac{A((N-1)t-2p\,t+p-x)}{t(N-1-p)}\bar{K}_p(x;t,N-1) - \frac{Bp(1-t)}{t(N-1-p)}\bar{K}_{p-1}(x;t;N-1) \qquad (3.37)$$
$$p = 1,2,\ldots,N-2$$

where

$$A = \sqrt{\frac{t(N-1-p)}{(1-t)(p+1)}},$$
$$B = \sqrt{\frac{t^2(N-1-p)(N-p)}{(1-t)^2(p+1)p}}$$

with

$$\bar{K}_0\,(x;t,N-1) = \sqrt{w(x;t,N-1)}$$
$$\bar{K}_1(x;t,N-1) = \left(1-\frac{x}{t(N-1)}\right)\left(\sqrt{\frac{t(N-1)}{1-t}}\right)\sqrt{w(x;t,N-1)}$$

Similarly, the following recursive relation can be used to compute the weight function given in Equation 3.33

$$w(x+1;t,N-1) = \left(\frac{N-1-x}{x+1}\right)\frac{t}{1-t}w(x;t,N-1) \qquad (3.38)$$

with $w(0;,t,N-1) = (1-t)^{N-1}$

The computation time of the Krawtchouk moments can be reduced significantly by using the symmetry property for the special case $t_1 = t_2 = 0.5$. The symmetry relation of the weighted Krawtchouk polynomial is given by [157]

$$\bar{K}_p(x;t,N-1) = (-1)^p\bar{K}_p(N-1-x;t,N-1) \qquad (3.39)$$

Using this symmetric relation for computing Krawtchouk moments of a 2-D image intensity function would save both computation time and storage space

for the Krawtchouk polynomials. For an image of size $N \times N$, where N is even, only the polynomials in the first quadrant, $0 \leq x, y \leq (N/2 - 1)$, need to be calculated. For an odd N, the image can be zero-padded to make N even.

The following matrix form of representation is useful for Krawtchouk moments computation in software packages such as MATLAB:

$$\mathbf{Q} = \mathbf{K_2 A K_1}^T \tag{3.40}$$

where $(.)^T$ denotes the transpose of the matrix and

$$\begin{aligned}
\mathbf{Q} &= \{Q_{ij}\}_{i,j=0}^{i,j=N-1} \\
\mathbf{K_v} &= \{\bar{K}_i(j; t_v, N-1)\}_{i,j=0}^{i,j=N-1} \\
\mathbf{A} &= \{f(i,j)\}_{i,j=0}^{i,j=N-1}
\end{aligned}$$

The image can be reconstructed by

$$\mathbf{A} = \mathbf{K_2}^T \mathbf{Q K_1} \tag{3.41}$$

Similar to the Tchebichef moments, Krawtchouk moments also do not have the problems mentioned in Subsection 3.2.1. Similar to the Tchebichef moments, the basis functions of Krawtchouk moments are orthogonal in the discrete domain of the image coordinate space. Therefore, implementation of these moments requires neither any discrete approximation nor coordinate space normalization. Appropriate scale factors are used for the weighted Krawtchouk polynomials to ensure numerical stability Therefore, implementation of Krawtchouk moments overcomes all the problems mentioned in Subsection 3.2.1 and found to be suitable as pattern feature in two-dimensional image analysis.

In all computations of Krawtchouk moments, the values of the parameters t_1 and t_2 are taken to be 0.5 because our region-of-interest is the whole image.

3.3 Printed Character Representation using Orthogonal Moments

In moment based character recognition systems, like other statistical pattern recognition systems, the image of a character is represented in terms of a set of d moments (features) of different order and is viewed as a point in a d-dimensional feature space [61]. Due to the variations in the images of a given character they generally map to more than one point in the space. The variations may be due to noise, font type, font size etc. Thus, the images of a character normally map to a cluster of points in the space. If the clusters formed by the images of different characters do not overlap with each other, they remain distinguishable. Choice of appropriate set of moments becomes important here. It is also important to keep the value of d minimum so that the computation involved is minimized. This section presents a study on representation of printed Assamese characters using Legendre, Tchebichef and Krawtchouk moments and gives a comparative analysis of their performance. The section also presents a study on use of the Principal Component Analysis and the Linear Discriminant Analysis for better representation.

3.3.1 The Representation Scheme

Let us consider an alphabet consisting n characters. From each character, k different noisy images are generated by adding Gaussian noise of zero mean and different standard deviations. These $N_s = n \times k$ noisy character images belong to n different classes where each class represents one character from the alphabet. Moments of order 0 to t are computed for each character image and then the images are represented in a $d = (t+1)(t+2)/2$ dimensional feature space where the features are the moments. The clusters formed by the images of the different characters are examined and if they (the clusters) do not overlap with each other then the representation scheme can effectively be used to recognize noisy characters of the alphabet.

Let, $\mathbf{X_{ri}}$ is the ith sample of the rth character ($r = 1, 2, \ldots, n;\ \ i = 1, 2, \ldots, k$) represented as a vector of d moments. Again let, $\mathcal{C}_r$ be the rth cluster formed by

the samples of the rth character. The centroid $\bar{\mathbf{X}}_\mathbf{r}$ of $\mathcal{C}_r$ is given by

$$\bar{\mathbf{X}}_\mathbf{r} = \frac{1}{k}\sum_{i=1}^{k} \mathbf{X_{ri}}, \quad r = 1, 2, \ldots, n; \;\; i = 1, 2, \ldots, k \tag{3.42}$$

In general, the clusters will have irregular shapes. However, for simplicity we model these clusters as hyperspheres. Now, let us consider the hypersphere in d dimensional Euclidean space centered at $\bar{\mathbf{X}}_\mathbf{r}$ corresponding to the rth cluster. The radius R_r of this hypersphere is the distance between the centroid and the sample which is farthest from the centroid and is given by

$$\begin{aligned} R_r = \max_i \{\sqrt{(\mathbf{X_{ri}} - \bar{\mathbf{X}}_\mathbf{r})'(\mathbf{X_{ri}} - \bar{\mathbf{X}}_\mathbf{r})}\}, \\ r = 1, 2, \ldots, n; \;\; i = 1, 2, \ldots, k \end{aligned} \tag{3.43}$$

These hyperspheres represent n clusters corresponding to the n characters of the alphabet. If these hyperspheres are non-overlapping, i.e. the clusters are disjoint then the noisy characters belonging to one class are clearly distinguishable from the other classes. Let, $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s})$ be the Euclidean distance between $\bar{\mathbf{X}}_\mathbf{r}$ and $\bar{\mathbf{X}}_\mathbf{s}$. For two clusters $\mathcal{C}_r$ and $\mathcal{C}_s$ centered at $\bar{\mathbf{X}}_\mathbf{r}$ and $\bar{\mathbf{X}}_\mathbf{s}$ respectively, we have the following three cases:

- $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) > R_r + R_s$ - the clusters are disjoint
- $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) = R_r + R_s$ - the clusters are touching each other
- $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) < R_r + R_s$ - the clusters are overlapping

Obviously, if $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) > R_r + R_s$, for all values of $r, s = 1, 2, \ldots, n$ and $r \neq s$, then all the clusters are disjoint.

To examine the disjointness of the clusters, the closest pair of clusters are considered and if the closest pair of clusters are disjoint, then all the clusters are disjoint. For this, let us denote the distance between two clusters $\mathcal{C}_r$ and $\mathcal{C}_s$ as $D(\mathcal{C}_r, \mathcal{C}_s)$ and defined as the difference between the Euclidean distance between the centroids of the clusters and the sum of their radii, i.e.,

$$D(\mathcal{C}_r, \mathcal{C}_s) = d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) - (R_r + R_s) \tag{3.44}$$

The distance given by the Equation 3.44 is based on Euclidean distance and therefore it will generally have a higher value at a higher dimension than a lower dimension. Since dimension d of one representation increases (or decreases) with moment order, therefore in order to compare the representations at different moment order, the distance measure given by Equation 3.44 is not a good measure of separability. For this purpose it is proposed to use a weighted distance D_w between the closest pair of clusters, where the weight is the inverse of the average of the radii of all clusters, i.e.,

$$\begin{aligned} D_w &= \frac{n}{\sum_{i=1}^{n} R_i} D(C_r, C_s) \\ &= \frac{n}{\sum_{i=1}^{n} R_i} \{d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s}) - (R_r + R_s)\} \end{aligned} \tag{3.45}$$

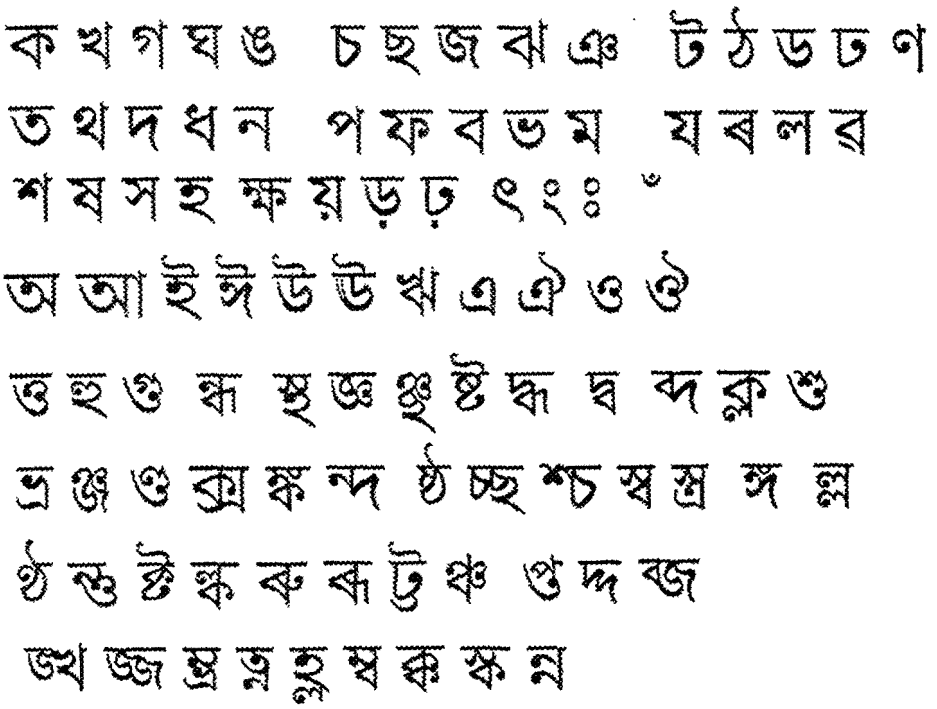


Figure 3.1: The Assamese characters used in the experiments (First three rows consonants, fourth row vowels and the last four rows conjunct characters

where C_r and C_s are the closest pair of clusters. A higher value of D_w implies better clustering. Clearly, $D_w < 0$, $D_w = 0$ and $D_w > 0$ mean overlapping, touching and disjoint clusters respectively.

3.3.2 Experimental Results

The alphabet for this experiment consists of 98 printed Assamese characters containing 41 consonants, 11 vowels and 46 conjunct characters (Yuktakshar) (Figure 3.1). The size of the character images varies from 30 × 30 to 60 × 60. For each character 100 noisy character images are generated by adding Gaussian noise of mean 0 and different standard deviations. These 9800 character images belong to 98 different classes where each class represents one character out of the 98 characters taken for this experiment.

After generating 9800 characters belonging to 98 classes, Legendre, Tchebichef and Krawtchouk moments of order from 0 to 30 are computed using Equations 3.12, 3.26 and 3.40 respectively, for each character image. Then the centroids $\bar{\mathbf{X}}_\mathbf{r}$ and the radii R_r of each class are computed using Equations 3.42 and 3.43 respectively. Then the values $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s})$ and $(R_r + R_s)$ are computed for all pairs of classes to find out the closest pair clusters and then the weighted distance between the closest pair of clusters D_w is computed using Equation 3.45 as explained in the previous subsection.

The experiment is repeated at different noise levels determined by the standard deviation Gaussian noise in the range of $5 - 50$ in steps of 5 and considering moment vectors of order $1, 2, \ldots, 30$. The experimental results of these representations using Legendre, Tchebichef and Krawtchouk moments are described below.

The recognition performance of the representation schemes on a dataset of 19600 noisy character images comprising 200 noisy images of each character of Figure 3.1 is also presented. The recognition is done using a minimum distance classifier. The Euclidean distances of the moment vector of an input image of an unknown character, from the centroids $\bar{\mathbf{X}}_\mathbf{r}$ of the clusters C_r, $r = 1, 2, \ldots, n$, are computed and the input image is assigned to the cluster of nearest cluster centroid and is recognized to be the character corresponding to that cluster.

Table 3.1: The values of D_w of the closest pair of clusters using Legendre moments at different noise level (σ)

Moment	Standard Deviation of Noise (σ)									
Order	5	10	15	20	25	30	35	40	45	50
1	0.09	-0.79	-1.09	-1.40	-1.66	-1.72	-1.64	-2.25	-1.94	-2.19
2	5.08	2.21	0.89	-0.17	-0.07	-0.36	-0.56	-0.78	-1.09	-1.35
3	4.28	1.57	0.50	-0.15	-0.27	-0.50	-0.69	-0.97	-0.89	-1.04
4	5.27	1.85	0.91	0.17	-0.37	-0.65	-0.69	-1.01	-1.05	-1.13
5	4.88	1.99	0.91	0.23	-0.28	-0.55	-0.79	-0.89	-0.99	-0.92
6	4.73	1.39	0.59	0.01	-0.54	-0.78	-1.06	-1.13	-1.13	-1.02
7	4.59	1.58	0.47	0.01	-0.50	-0.74	-0.93	-1.21	-1.09	-1.10
8	4.95	1.75	0.59	-0.09	-0.39	-0.63	-0.74	-0.95	-1.05	-1.27
9	4.99	1.78	0.59	0.02	-0.34	-0.60	-0.89	-1.02	-1.04	-1.09
10	6.01	2.15	0.98	0.31	-0.27	-0.55	-0.70	-0.88	-0.98	-1.04
11	6.53	2.60	1.16	0.52	-0.04	-0.36	-0.61	-0.73	-0.79	-0.97
12	6.60	2.69	1.22	0.49	-0.03	-0.35	-0.56	-0.78	-0.77	-0.96
13	7.41	3.04	1.53	0.71	0.13	-0.22	-0.46	-0.65	-0.69	-0.85
14	6.84	2.79	1.34	0.58	0.06	-0.26	-0.49	-0.67	-0.79	-0.95
15	7.34	3.02	1.41	0.68	0.17	-0.17	-0.45	-0.58	-0.71	-0.95
16	7.66	3.25	1.54	0.79	0.21	-0.15	-0.35	-0.55	-0.67	-0.91
17	8.31	3.57	1.82	1.02	0.31	-0.07	-0.25	-0.44	-0.56	-0.85
18	9.87	4.41	2.36	1.43	0.74	0.30	0.00	-0.25	-0.43	-0.60
19	9.74	4.37	2.31	1.38	0.67	0.24	-0.02	-0.24	-0.45	-0.64
20	11.07	5.08	2.80	1.73	1.00	0.52	0.22	-0.07	-0.29	-0.46
21	10.90	4.96	2.76	1.68	0.98	0.50	0.20	-0.07	-0.32	-0.47
22	11.08	5.09	2.87	1.72	1.01	0.52	0.18	-0.04	-0.30	-0.47
23	10.89	5.00	2.80	1.65	0.89	0.41	0.17	-0.08	-0.32	-0.49
24	10.82	4.94	2.68	1.63	0.91	0.43	0.12	-0.11	-0.34	-0.51
25	10.69	4.90	2.70	1.60	0.90	0.43	0.11	-0.15	-0.38	-0.49
26	11.03	5.06	2.82	1.72	0.98	0.49	0.17	-0.06	-0.32	-0.46
27	10.92	4.99	2.76	1.71	0.96	0.48	0.15	-0.12	-0.36	-0.47
28	11.24	5.13	2.92	1.81	1.04	0.55	0.23	-0.05	-0.31	-0.45
29	11.14	5.08	2.90	1.78	1.04	0.56	0.20	-0.06	-0.33	-0.50
30	11.14	5.10	2.89	1.79	1.03	0.54	0.19	-0.07	-0.36	-0.49

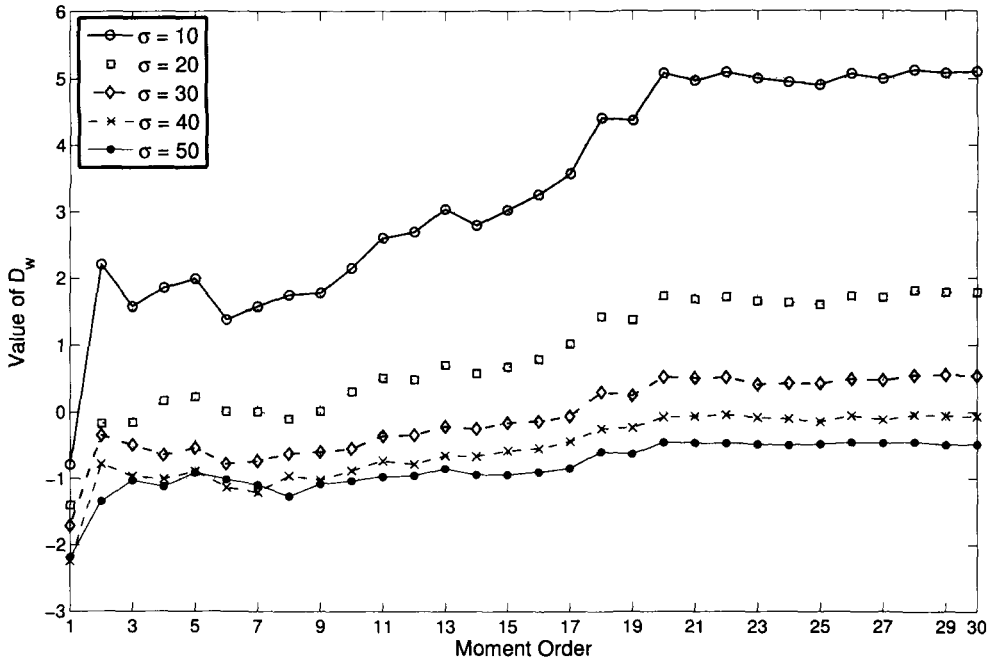


Figure 3.2: The value of D_w using Legendre moments at different noise levels

3.3.2.1 Legendre Moments

In this subsection the experimental results using orthogonal Legendre moments are presented.

Table 3.1 shows the values of D_w for different moment order at different noise level(σ). It can be seen from the table that when standard deviation of noise, $\sigma = 5$, Legendre moments up to order 1 are sufficient to represent the character images in order to make all the clusters disjoint. Since that dataset consists of noisy images of a single font-type and single font-size only, for small random variations ($\sigma = 5$) disjoint clusters are observed even at Legendre moments of order = 1. When σ is increased to 10 and the representation is done with moments up to order 2 or above, all the clusters become disjoint. Similarly, it can be observed from the columns corresponding to other values of σ that number of moments required for representation increases along with noise level to make the clusters disjoint. At a noise level $\sigma = 35$, moments up to order 20 or above are

necessary to represent the character images. For a noise level $\sigma \geq 45$, moments up to order 30 are not sufficient to represent the character images.

The plots of Figure 3.2 show the values of D_w for Legendre moments of order $1, 2, \ldots, 30$ at noise levels $\sigma = 10, 20, 30, 40, 50$. From the upward trend of all the plots in this figure (Figure 3.2), it is clear that the representation accuracy improves along with the order of moments considered for representation. As expected, the representation accuracy reduces when noise level increases.

The recognition accuracies of the representation scheme at noise levels $\sigma = 5, 10$ and 15 are found to be 100% using Legendre moments of order 2 and above. As expected, higher order moments are required to achieve 100% recognition accuracy at higher noise level. For example, at noise level $\sigma = 40$, 100% recognition accuracy is acheived using Legendre moments of order 7 and above. It is to be noted that the values of D_w are computed from two nearest cluster pairs. A negative value of D_w indicates that there are some overlapping clusters and recognition errors may occur only when an input character is mapped to such an overlapping region. Therefore, 100% recognition accuracies are obtained, in many cases, even when the values of D_w are negative. For example, when noise level is $\sigma = 40$, though the value of D_w is -1.21, 100% recognition acuracy is obtained at order 7.

3.3.2.2 Tchebichef Moments

The experimental results using discrete orthogonal Tchebichef moments are presented in Table 3.2 and Figure 3.3.

Table 3.2 shows the values of D_w for different moment order at different values of σ. It can be seen from Table 3.2 that the representation performance decreases with increase in noise levels. As in the case of Legendre moments, Tchebichef moments up to order 1 are sufficient to distinctly represent the character images at $\sigma = 5$. But unlike Legendre moments, Tchebichef moments up to order 2 are sufficient to have disjoint clusters for noise levels (σ) from 10 to 25. At $\sigma = 35$ and $\sigma = 40$, the maximum moment orders required for disjoint clusters are 16 and 22 respectively.

Figure 3.3 shows the plots of D_w at noise levels $\sigma = 10, 20, 30, 40, 50$ with

Table 3.2: The values of D_w of the closest pair of clusters using Tchebichef moments at different noise level (σ)

Moment	Standard Deviation of Noise (σ)									
Order	5	10	15	20	25	30	35	40	45	50
1	0.38	-0.96	-1.39	-1.39	-1.40	-1.62	-1.55	-1.75	-1.79	-1.96
2	8.14	3.58	1.81	0.87	0.45	-0.07	-0.35	-0.70	-0.80	-1.10
3	6.97	3.00	1.43	0.67	0.31	0.02	-0.35	-0.57	-0.56	-0.80
4	6.99	3.03	1.50	0.73	0.37	0.03	-0.36	-0.53	-0.59	-0.79
5	6.91	2.95	1.53	0.80	0.25	-0.01	-0.37	-0.42	-0.59	-0.79
6	6.69	2.82	1.44	0.68	0.19	0.01	-0.39	-0.61	-0.68	-0.80
7	8.07	3.40	1.53	0.66	0.13	-0.08	-0.48	-0.51	-0.74	-0.82
8	7.96	3.32	1.55	0.67	0.16	-0.16	-0.48	-0.52	-0.87	-0.88
9	7.97	3.27	1.54	0.70	0.18	-0.11	-0.42	-0.51	-0.82	-0.92
10	8.10	3.38	1.63	0.75	0.27	-0.09	-0.40	-0.51	-0.78	-0.92
11	8.76	3.76	1.86	0.92	0.42	0.08	-0.25	-0.38	-0.72	-0.79
12	8.59	3.68	1.84	0.88	0.34	0.05	-0.29	-0.40	-0.72	-0.78
13	8.85	3.86	1.99	0.99	0.44	0.11	-0.19	-0.38	-0.70	-0.77
14	8.73	3.78	1.98	0.96	0.43	0.10	-0.24	-0.42	-0.70	-0.77
15	9.36	4.13	2.20	1.17	0.62	0.23	-0.12	-0.34	-0.59	-0.70
16	9.94	4.48	2.42	1.37	0.74	0.36	0.00	-0.25	-0.49	-0.63
17	10.25	4.63	2.53	1.45	0.82	0.41	0.06	-0.20	-0.43	-0.57
18	10.96	5.03	2.82	1.68	1.01	0.54	0.16	-0.11	-0.31	-0.45
19	11.22	5.18	2.91	1.78	1.07	0.60	0.21	-0.11	-0.25	-0.44
20	11.58	5.36	3.10	1.90	1.15	0.68	0.28	-0.04	-0.23	-0.39
21	11.67	5.42	3.11	1.94	1.14	0.69	0.31	-0.02	-0.20	-0.38
22	11.98	5.57	3.24	2.03	1.22	0.74	0.35	0.03	-0.16	-0.35
23	11.92	5.53	3.21	1.98	1.18	0.74	0.33	0.04	-0.17	-0.36
24	12.00	5.57	3.24	2.02	1.19	0.77	0.35	0.05	-0.15	-0.34
25	11.99	5.57	3.24	2.00	1.20	0.77	0.36	0.03	-0.16	-0.34
26	12.09	5.61	3.27	2.04	1.24	0.79	0.36	0.06	-0.14	-0.33
27	12.16	5.64	3.30	2.05	1.26	0.79	0.37	0.06	-0.14	-0.34
28	12.21	5.68	3.31	2.05	1.28	0.79	0.39	0.07	-0.13	-0.34
29	12.11	5.63	3.27	2.01	1.25	0.76	0.36	0.06	-0.15	-0.34
30	12.12	5.64	3.28	1.99	1.25	0.76	0.36	0.06	-0.15	-0.35

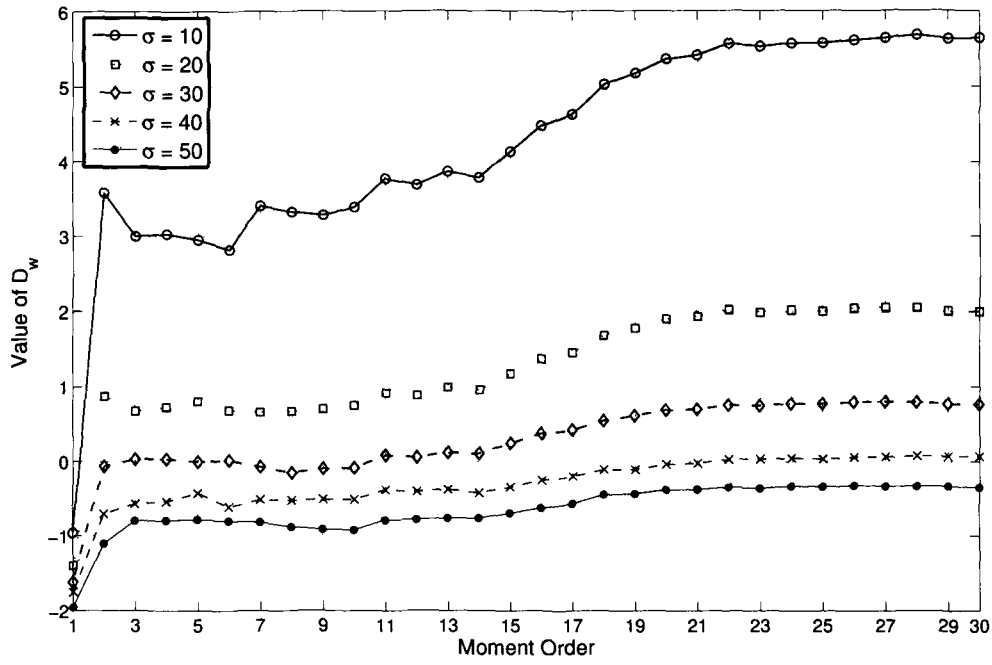


Figure 3.3: The value of D_w using Tchebichef moments at different noise levels

varying moment orders. Like the plots in Figure 3.2, here also the plots show upward trend indicating the improved representation accuracy with increasing moment order.

The recognition accuracies of the representation scheme at noise levels $\sigma = 5$ to $\sigma = 25$ are found to be 100% using Tchebichef moments of order 2 and above. As expected, in case Tchebichef moments also, higher order moments are required to achieve 100% recognition accuracy at higher noise level. For example, at noise level $\sigma = 30$, 100% recognition accuracy is acheived using Legendre moments of order 3 and above. In case of Tchebichef moments also, like Legendre moments, 100% recognition accuracies are obtained, in many cases, even when the values of D_w are negative. For example, when noise level is $\sigma = 40$, though the value of D_w is -0.61, 100% recognition acuracy is obtained at order 6.

3.3.2.3 Krawtchouk Moments

The experimental results using discrete orthogonal Krawtchouk moments are presented in Table 3.3 and Figure 3.4. Table 3.3 shows the values of D_w for different moment order at different noise levels.

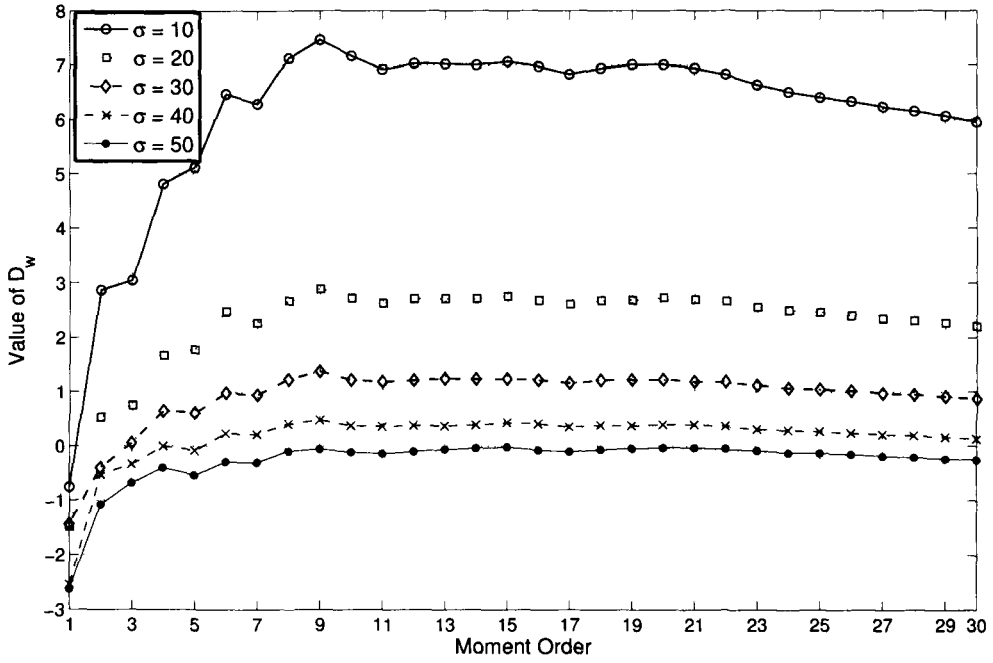


Figure 3.4: The value of D_w using Krawtchouk moments at different noise levels

It can be seen from Table 3.2 that when the noise level is $\sigma = 5$, like Legendre and Tchebichef moments, the cluster are disjoint when the characters images are represented by Krawtchouk moments up to order 1. The clusters remain disjoint up to a noise level $\sigma = 20$ using Krawtchouk moments up to order 2 and use of moments up to order 3 can make the clusters disjoint at $\sigma = 25$ and $\sigma = 30$. Similarly, at $\sigma = 35$ and $\sigma = 40$ all clusters are disjoint using moments up to order 4 or above and 6 or above respectively.

Figure 3.4 shows the plots of D_w at noise levels $\sigma = 10, 20, 30, 40, 50$ with varying moment order. Unlike the plots in Figure 3.2 and Figure 3.3, the plots of this figure show upward trend up to moment order 9, then remains

Table 3.3: The values of D_w of the closest pair of clusters using Krawtchouk moments at different noise level (σ)

Moment	Standard Deviation of Noise (σ)									
Order	5	10	15	20	25	30	35	40	45	50
1	0.60	-0.76	-1.06	-1.49	-1.55	-1.45	-1.75	-2.54	-2.68	-2.63
2	7.68	2.87	1.36	0.53	-0.04	-0.41	-0.55	-0.52	-0.77	-1.09
3	7.95	3.05	1.60	0.75	0.18	0.04	-0.25	-0.34	-0.47	-0.69
4	11.45	4.81	2.84	1.67	0.95	0.63	0.21	-0.01	-0.19	-0.41
5	12.27	5.11	2.92	1.78	1.10	0.60	0.21	-0.08	-0.25	-0.54
6	14.74	6.47	3.89	2.48	1.56	0.97	0.65	0.23	0.06	-0.30
7	14.14	6.29	3.80	2.27	1.51	0.94	0.61	0.22	0.00	-0.31
8	15.50	7.12	4.31	2.68	1.83	1.22	0.80	0.41	0.08	-0.10
9	16.18	7.46	4.39	2.90	1.97	1.37	0.91	0.48	0.24	-0.07
10	15.50	7.16	4.14	2.73	1.84	1.21	0.78	0.37	0.20	-0.12
11	14.99	6.91	3.93	2.63	1.72	1.17	0.70	0.36	0.07	-0.15
12	15.23	7.03	4.07	2.72	1.78	1.21	0.77	0.38	0.11	-0.11
13	15.12	7.02	4.03	2.71	1.81	1.23	0.80	0.37	0.14	-0.07
14	15.02	7.00	4.03	2.72	1.83	1.22	0.76	0.38	0.13	-0.05
15	15.16	7.06	4.13	2.75	1.84	1.22	0.82	0.43	0.16	-0.03
16	14.91	6.97	4.07	2.69	1.82	1.21	0.78	0.41	0.14	-0.09
17	14.62	6.82	3.97	2.62	1.78	1.15	0.73	0.35	0.12	-0.11
18	14.82	6.93	4.06	2.68	1.83	1.20	0.76	0.38	0.16	-0.08
19	14.97	7.00	4.11	2.69	1.85	1.21	0.80	0.37	0.18	-0.06
20	14.97	7.00	4.16	2.73	1.85	1.21	0.81	0.39	0.19	-0.04
21	14.84	6.94	4.11	2.70	1.80	1.18	0.75	0.40	0.18	-0.04
22	14.64	6.82	4.05	2.66	1.78	1.17	0.71	0.36	0.15	-0.06
23	14.30	6.63	3.93	2.56	1.69	1.11	0.66	0.31	0.11	-0.09
24	14.03	6.50	3.84	2.50	1.63	1.06	0.63	0.29	0.08	-0.14
25	13.86	6.40	3.77	2.46	1.60	1.04	0.60	0.26	0.05	-0.14
26	13.69	6.33	3.73	2.41	1.55	1.01	0.58	0.24	0.01	-0.16
27	13.48	6.23	3.66	2.35	1.52	0.96	0.54	0.21	-0.02	-0.19
28	13.32	6.15	3.61	2.31	1.48	0.94	0.52	0.19	-0.03	-0.21
29	13.12	6.07	3.55	2.27	1.45	0.91	0.49	0.16	-0.07	-0.24
30	12.91	5.95	3.48	2.21	1.40	0.87	0.46	0.13	-0.10	-0.26

steady from order 10 to around order 20 and beyond order 20 the values of D_w show a decreasing tendency. This is because though the representation accuracy increases initially with increasing moment order, at some point, the inaccuracy due to noise of the higher order moments outdo the added accuracy of them. It is also observed from the plots of Figure 3.4 that the representation accuracy reduces with increasing noise level.

The recognition accuracies of the representation scheme at noise levels $\sigma = 5$ to $\sigma = 30$ are found to be 100% using Krawtchouk moments of order 2 and above. In case of Krawtchouk moments also, higher order moments are required to achieve 100% recognition accuracy at higher noise level. For example, at noise level $\sigma = 40$, 100% recognition accuracy is acheived using Legendre moments of order 4 and above. In case of Krawtchouk moments also, like Legendre and Tchebichef moments, 100% recognition accuracies are obtained, in many cases, even when the values of D_w are negative. For example, when noise level is $\sigma = 40$, though the value of D_w is -0.34, 100% recognition acuracy is obtained at order 4.

3.3.2.4 Comparison among Legendre, Tchebichef and Krawtchouk Moments

From the Figures 3.2, 3.3 and 3.4 and also from the Tables 3.1, 3.2 and 3.3 it can be observed that, in general, the performance off all the moments show upward trends initially with increasing moment order. The values of D_w for Legendre and Tchebichef moments maintain this upward trend up to moment order 30 while that for Krawtchouk moments show downward trend beyond order 20. In order to have a better view of the comparative performance among these three moments for noisy character representation, the values of D_w for all these three moments are plotted at noise level $\sigma = 10$ and 30 in Figure 3.5, and 3.6 respectively. From these two figures, it can be observed that Krawtchouk moments show the best performance among the three moments particularly at low moment order. Though the values of D_w starts decreasing beyond moment order 20, they are above the corresponding values for the other two moment up to order 30. Performance of Tchebichef moments is better than that of Legendre

moments as indicated by the values of D_w for all moment orders considered in the experiment. Similar results are obtained for other noise levels also.

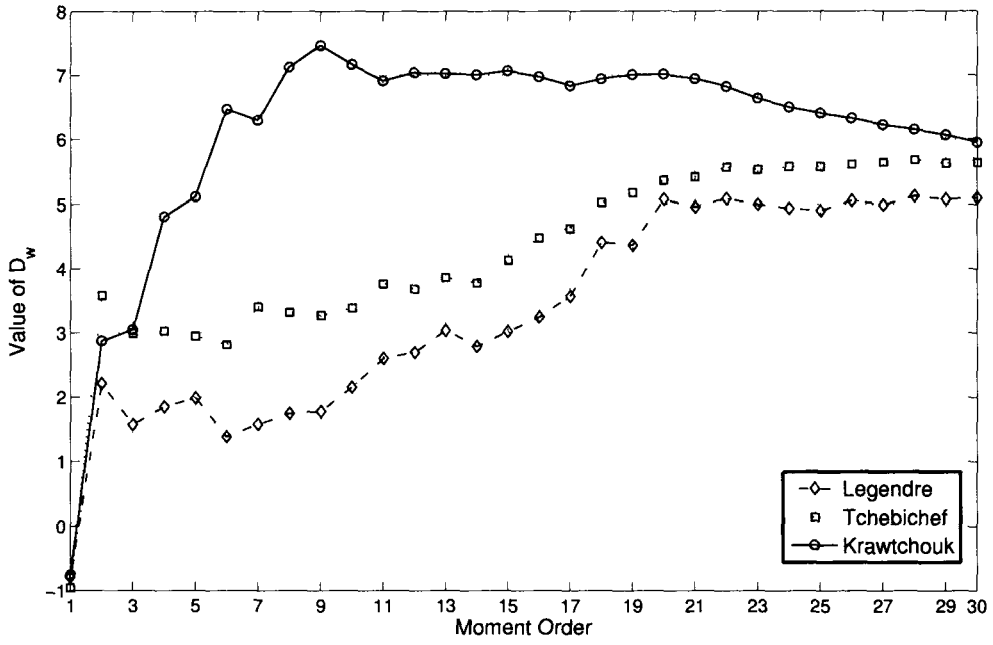


Figure 3.5: The value of D_w using Legendre, Tchebichef and Krawtchouk moments at noise level $\sigma = 10$

The table 3.4 presents a summary view of the performance of the three moments. The first column is the noise level added to the character images. The next three columns show the lowest moment order at which the clusters become disjoint for Legendre, Tchebichef and Krawtchouk moments respectively. The fifth, sixth and seventh columns show the maximum values of D_w and the remaining three columns show the moment orders at which D_w have the maximum values for Legendre, Tchebichef and Krawtchouk moments respectively. It can be observed from this table that Krawtchouk moment-based method require considerably smaller number of moments to have disjoint clusters followed by Tchebichef moment-based method. It is also seen that values of D_ws obtained from Krawtchouk moments are the highest among all the three moments. The second highest values of D_ws are obtained from Tchebichef moments. Finally, from the last three columns it is observed that Krawtchouk moments

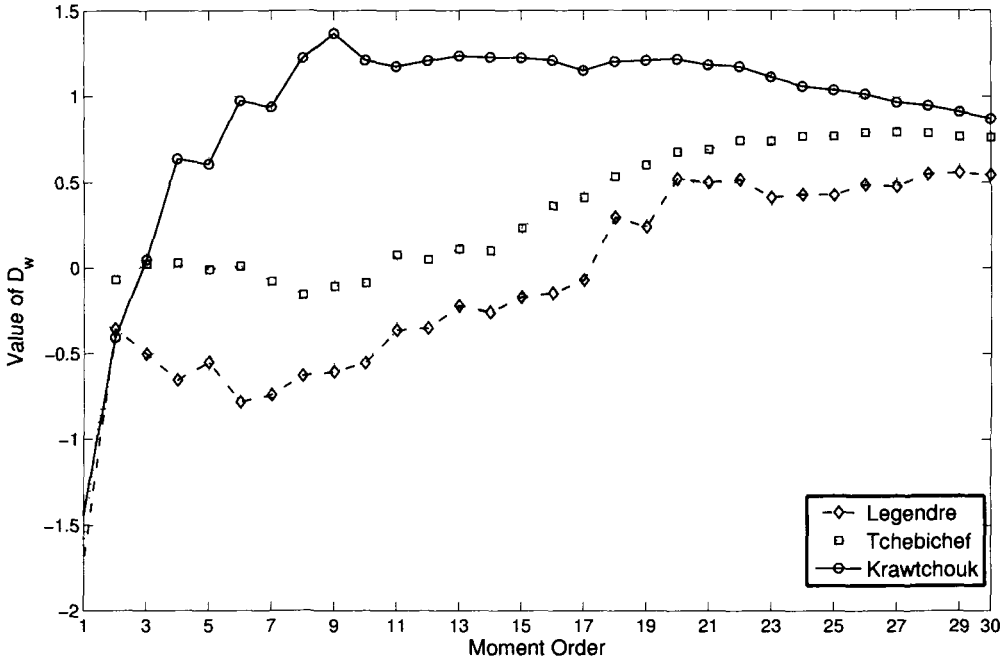


Figure 3.6: The value of D_w using Legendre, Tchebichef and Krawtchouk moments at noise level $\sigma = 30$

give best representation (maximum values of D_w) at significantly low order of moments whereas the other two moments give their best representation nearly at same moment order. From these observations, it can be concluded that performance of Krawtchouk moments is the best followed by Tchebichef moments for representation of printed Assamese characters.

The recognition performances of all the three moments show upward trends with increasing moment order. Higher order moments are required at higher noise levels for all the three moments. Among these three moments, Krawtchouk moment-based representations give the best performances, as they generally require less number of moments particularly higher noise levels, followed by Tchebichef moments.

Table 3.4: Comparison of the values of D_w computed from Legendre(L) Tchebichef(T) and Krawtchouk(K) moments

Noise level (σ)	Lowest moment order for disjoint clusters			Maximum value of D_w			Moment order at maximum D_w		
	L	T	K	L	T	K	L	T	K
10	2	2	2	5.13	5.68	7.46	28	28	9
20	4	2	2	1.81	2.05	2.90	28	27	9
30	18	11	3	0.56	0.79	1.37	29	27	9
40	-	22	6	-0.04	0.07	0.48	22	28	9
50	-	-	-	-0.45	-0.33	-0.03	28	26	15

3.4 Principal Component Analysis for Improved Representation

The principal component analysis (PCA) is a well-known technique for dimensionality reduction. It finds a projection which best represents the data in least-squares sense [33]. As the data contain less redundancy because of orthogonality of the moments, PCA is applied here only to find out the major dimensions corresponding to the variation of the characters of the alphabet and discarding those corresponding to the minor variation due to noise. Since the data consists of noisy character images of n characters of the alphabet, there are at most $n-1$ major dimensions and the remaining dimensions are due to noise. Therefore, if the data consisting of the moment vectors of the noisy characters images, projected on the first $n-1$ principal components according to their decreasing eigen values, is expected to give the best representation of the character images which is also supported by the experimental results to a great extent. In this section, a study on use of PCA of moment vectors for improved representation of printed Assamese characters is presented. The theory of Principal Component Analysis and the experimental results are presented in the following two subsections.

3.4.1 Principal Component Analysis

The Principal Component Analysis (PCA) is a simple, non-parametric method of extracting relevant information from confusing data sets. Applications of PCA are found in all forms of analysis - from neuroscience to computer graphics.

Let, $\mathbf{X}$ is an $N_s \times d$ data matrix where N_s is the number of samples and d is the number of attributes. The goal of PCA of this data matrix may be summarized as [136]:

> find some orthonormal matrix $\mathbf{P}$ where $\mathbf{Y} = \mathbf{P}\mathbf{X}^T$ such that $\mathbf{C_Y} = \frac{1}{N_s-1}\mathbf{Y}\mathbf{Y}^T$ is diagonalized. The rows of $\mathbf{P}$ are the principal components of $\mathbf{X}$

.

Let us express $\mathbf{C_Y}$ in terms of $\mathbf{P}$

$$\begin{aligned}
\mathbf{C_Y} &= \frac{1}{N_s - 1}\mathbf{Y}\mathbf{Y}^T \\
&= \frac{1}{N_s - 1}(\mathbf{P}\mathbf{X}^T)(\mathbf{P}\mathbf{X}^T)^T \\
&= \frac{1}{N_s - 1}\mathbf{P}\mathbf{X}^T\mathbf{X}\mathbf{P}^T \\
&= \frac{1}{N_s - 1}\mathbf{P}(\mathbf{X}^T\mathbf{X})\mathbf{P}^T \\
\mathbf{C_Y} &= \frac{1}{N_s - 1}\mathbf{P}\mathbf{A}\mathbf{P}^T
\end{aligned}$$

where $\mathbf{A} \equiv \mathbf{X}^T\mathbf{X}$ is a symmetric matrix. A symmetric matrix is diagonalized by an orthogonal matrix of its eigenvectors. That is

$$\mathbf{A} = \mathbf{E}\mathbf{D}\mathbf{E}^T \tag{3.46}$$

where $\mathbf{D}$ is a diagonal matrix and $\mathbf{E}$ is a matrix of eigenvectors of $\mathbf{A}$ arranged as columns. Taking $\mathbf{P}$ to be a matrix where each row $\mathbf{p_i}$ is an eigenvector of $\mathbf{X}^T\mathbf{X}$, we get $\mathbf{P} \equiv \mathbf{E}^T$. Substituting this into Equation 3.46, we find $\mathbf{A} = \mathbf{P}^T\mathbf{D}\mathbf{P}$. Now,

It is evident that $\mathbf{P} \equiv \mathbf{E}^T$ diagonalizes $\mathbf{C_Y}$, which was the goal of PCA. The above results can be summarized as follows:

- The principal components (PC) of $\mathbf{X}$ are the eigenvectors of $\mathbf{X}^T\mathbf{X}$; i.e the rows of $\mathbf{P}$.
- The ith diagonal value of $\mathbf{C_Y}$ is the variance of $\mathbf{X}$ along $\mathbf{p_i}$.

In practice, computing of PCA of a data set $\mathbf{X}$ consists of:

1. Subtracting off the mean of each attributes and
2. Computing the eigenvectors of $\mathbf{X}^T\mathbf{X}$.

3.4.2 Experimental Results

The experiments are performed on the same dataset used in the experiments presented in Section 3.3. The principal components (PC) of the moment vectors containing moments of order 0 to 30 are obtained using 'princomp' function of MATLAB. The character images are represented using a representation scheme similar to the scheme presented in 3.3 where the moment vectors are replaced by the principal component scores. The principal component scores are used to compute the cluster centroids, the radii, the distances between the clusters and finally the weighted distances between the closest pair of clusters D_w. The results of these experiments are presented in the remaining part of this subsection.

The recognition performance of the representation schemes, on the same dataset used in Subsection 3.3.2 of 19600 noisy character images of 98 characters of Figure 3.1, is also presented. The recognition is done using a minimum distance classifier. The moment vector of an input image of an unknown character is transformed by the matrix containing the principal components of the dataset used in representation. Euclidean distances of this transformed vector from the centroids $\bar{\mathbf{X}}_\mathbf{r}$ of the clusters C_r $(r = 1, 2, \ldots, n)$ are computed and the input image is assigned to the cluster of nearest cluster centroid and is recognized to be the character corresponding to that cluster.

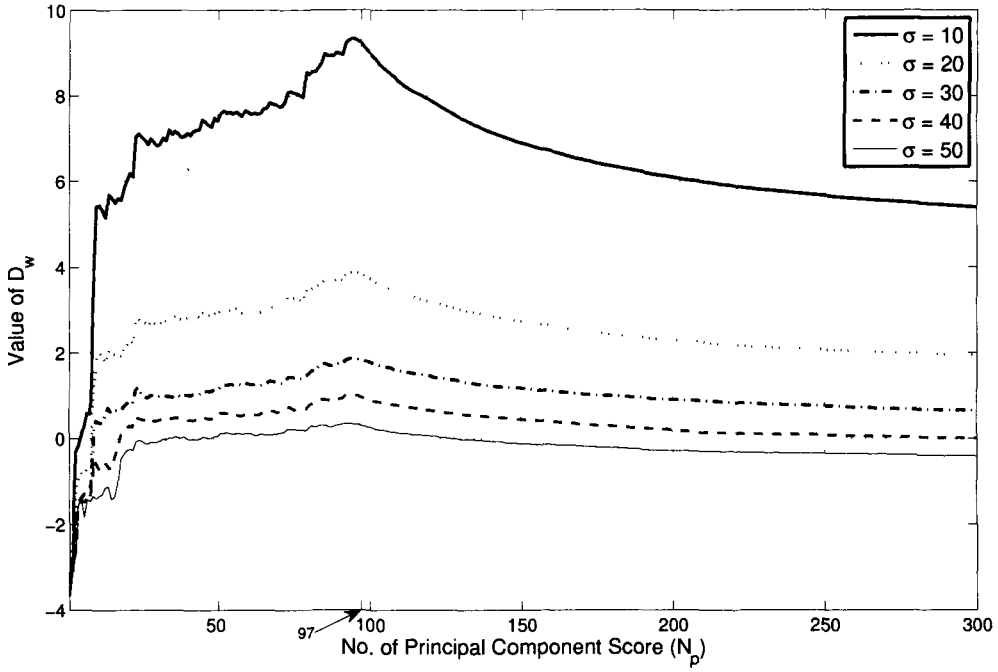


Figure 3.7: The value of D_w using PC scores of Legendre moments

3.4.2.1 Principal Components of Legendre Moments

Figure 3.7 shows the plots of D_w computed from different numbers of principal component scores (N_p) of Legendre moments at noise levels $\sigma = 10, 20, 30, 40$ and 50. It can be observed from these plots that the D_ws obtain maximum values near $N_p = 97$. Though the maximum values of D_w are not exactly at $N_p = 97$, the values of D_ws at $N_p = 97$ are very close to the maximum values of D_w (the maximum value of D_w are at most 1.15 times of its value at $N_p = 97$).

A comparison of the plots of Figure 3.7 with that of Figure 3.2 shows that considerable improvement can be achieved by representing the character images using principal component scores of the Legendre moments instead of the moments themselves. The Table 3.5 gives a summarized view of the comparative performance of these two approaches. The first column of this table shows the standard deviation (σ) of Gaussian noise added to the character images. The values in the second column are the maximum values of D_ws from the Legendre

Table 3.5: The maximum values of D_w computed from Legendre moments and principal component scores of the Legendre moments

Noise level (σ)	Legendre Moment			Principal Component(PC) Score		
	Maximum D_w	Moment order	No. of moments	Maximum D_w	No. of PC scores (N_p)	
10	5.13	28	435	9.34	94	10
20	1.81	28	435	3.88	95	10
30	0.56	29	465	1.85	94	14
40	-0.04	22	276	1.01	93	17
50	-0.45	28	435	0.34	93	19

moments. The third and fourth columns give the moment order and number of moments respectively that give the maximum values of D_ws. The fifth and sixth columns indicate the maximum values of D_w and number of principal component scores that give these maximum values respectively. The last column in this table gives the minimum number of principal component scores that give a value of D_w which is higher than or at least equal to the maximum value of D_w given by the Legendre moments. For example, from the first row of this table it is observed that, when the standard deviation of noise $\sigma = 10$, then the maximum value of D_w from Legendre moments is 5.13 which is obtained when the images are represented by Legendre moments of order 0 to 28, that is 435 moments (one moment of order 0, two moments of order 1 and so on). Whereas, the maximum values of D_ws obtained from principal component scores of the Legendre moments is 9.34 which is obtained when representation is done with 94 principal component scores. And finally, the value 10 in the last column indicates that a representation with 10 principal component scores gives a value of $D_w \geq 5.13$. It is observed that the maximum value of D_w obtained from the principal component scores are much higher than those obtained from the Legendre moments themselves. Also, the values of D_w obtained from principal component scores in lower dimensional representations are as good as those obtained from Legendre moments in much higher dimensions. For example, at noise level $\sigma = 10$, the value of D_w obtained from 10 principal component scores (10 dimensional Euclidean space) is as good

as that obtained from 435 Legendre moments (435 dimensional Euclidean space). Similar results are also observed at other noise levels.

A summary of recognition results of Principal Component Analysis-based representations are presented in Table 3.6. From this table in can be observed that, PCA-based representations are more efficient as they give same performance at lower dimensions in comparison to the moments themselves. For example, minimum number of Legendre moments needed for 100% recognition accuracy at noise level $\sigma = 30$ is 15 (order 4), whereas a minimum of 9 PC scores of Legendre moments give the same recognition accuracy at that noise level.

Table 3.6: Minimum number of moments or Principal Component(PC) Scores needed for 100% recognition accuracy

Noise	Legendre		Tchebichef		Krawtchouk	
level (σ)	Moments	PC scores	Moments	PC scores	moments	PC scores
10	6	4	6	3	6	3
20	10	6	6	3	6	4
30	15	9	10	4	6	4
40	36	10	28	6	15	5
50	66	14	45	8	28	7

3.4.2.2 Principal Components of Tchebichef Moments

Figure 3.8 shows the plots of D_w, computed from principal component scores of Tchebichef moments, against the numbers of principal component scores (N_p) at noise levels $\sigma = 10, 20, 30, 40$ and 50. Unlike Legendre moments, it is observed from these plots that the D_w obtain the maximum value exactly at $N_p = 97$ for noise levels $\sigma = 10, 20, 30$ and 40. On the other hand, at noise level $\sigma = 50$, the value of D_w at $N_p = 97$ is very close to the maximum value of D_w (the maximum value of D_w is only 1.0061 times the value of D_w at $N_p = 97$).

A comparison of the plots of Figures 3.8 and 3.3 shows that, like Legendre moments, significant improvement can be achieved by representing the character images using principal component scores of the Tchebichef moments instead

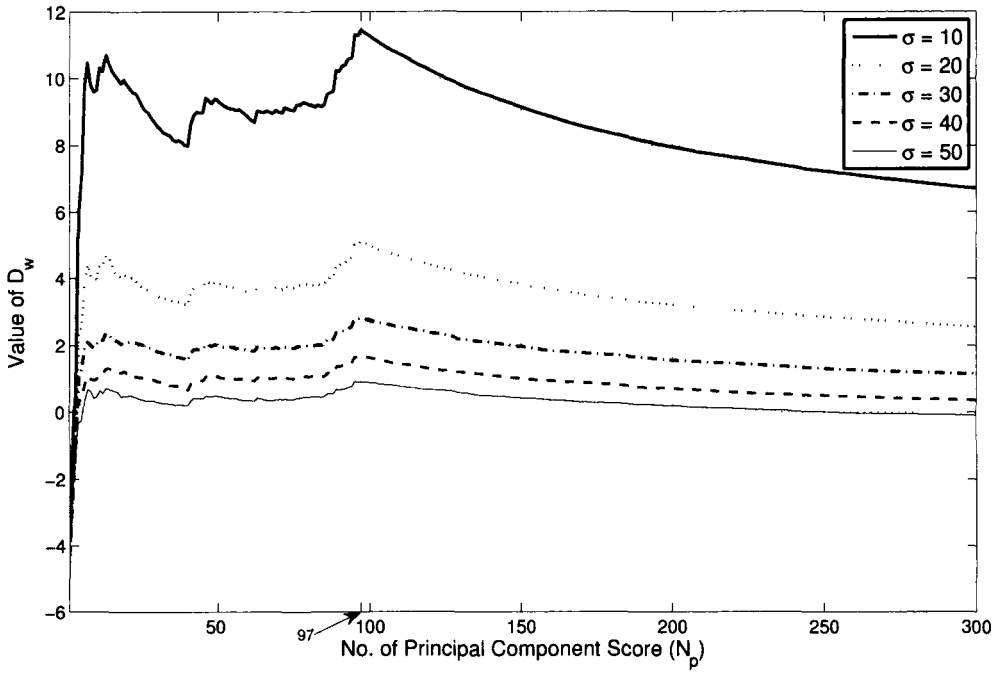


Figure 3.8: The value of D_w using PC scores of Tchebichef moments

of the moments themselves. Table 3.7 provides a summarized view of the comparative performance of the representations using Tchebichef moments and principal component scores of these moments. From this table, it is found that the maximum values obtained from principal component scores of Tchebichef moments are much higher than those obtained from the moments themselves. For example, the maximum values of D_ws, when $\sigma = 10$, for the Tchebichef moments and the principal component scores of the Tchebichef moments are 5.68 and 11.45 respectively. The corresponding dimensions of representations are 435 and 97 for Tchebichef moments and principal component scores of Tchebichef moments respectively. Also from the last column it is found that, the value of D_w obtained from 4 principal component scores of Tchebichef moments are at least equal to the value of D_w obtained from 435 Tchebichef moments (at noise level $\sigma = 10$). Similar results are also observed at all other noise levels.

A summary of recognition results of Principal Component Analysis-based representations using Tchebichef moments are also given in Table 3.6. From this

Table 3.7: The maximum values of D_w computed from Tchebichef moments and principal component scores of the Tchebichef moments

Noise level (σ)	Tchebichef Moment			Principal Component(PC) Score		
	Maximum D_w	Moment order	No. of moments	Maximum D_w	No. of PC score (N_p)	
10	5.68	28	435	11.45	97	4
20	2.05	27	406	5.09	97	5
30	0.79	27	406	2.79	97	4
40	0.07	28	435	1.67	97	4
50	-0.33	26	378	0.89	95	2

table in can be observed that, PCA-based representations are more efficient as they give same performance at lower dimensions in comparison to the moments themselves. For example, minimum number of Tchebichef moments needed for 100% recognition accuracy at noise level $\sigma = 30$ is 10 (order 3), whereas a minimum of 4 PC scores of Tchebichef moments are sufficient to get the same recognition accuracy at that noise level.

3.4.2.3 Principal Components of Krawtchouk Moments

Similar to the experiments done using Legendre and Tchebichef moments, principal component scores of Krawtchouk moments are used to compute the values of D_ws. Figure 3.9 shows the plots of these values against number of principal component scores (N_p) at noise levels $\sigma = 10, 20, 30, 40$ and 50. It is observed from these plots that the D_w obtain the maximum value at $N_p = 96$ at all noise levels. These maximum values are at most 1.0097 times of the value of D_w at $N_p = 97$.

Similar to the Tables 3.5 and 3.7, Table 3.8 summarizes the comparative performance of Krawtchouk moment-based representation and principal component scores of Krawtchouk moment-based representation. It is observed, from this table, that the maximum values of D_w obtained from the principal component scores are much higher than those obtained from the moments

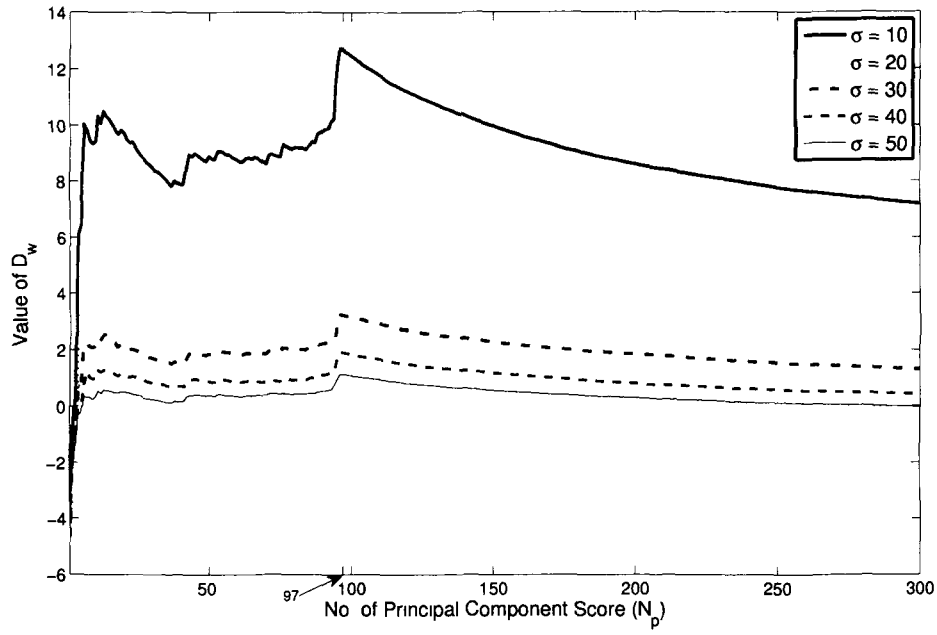


Figure 3.9: The value of D_w using PC scores of Krawtchouk moments

themselves. But, unlike Legendre and Tchebichef moments, the maximum values of D_w, based on Krawtchouk moments, are obtained comparatively at lower dimensions than the maximum values of D_w based on principal component scores of Krawtchouk moments. Though the maximum values of D_w, based on principal component scores of Krawtchouk moments, are obtained in a higher dimension than those obtained from the Krawtchouk moments in most cases, the clusters are more distinct in the former case as indicated by higher values of D_w. Also, form the last column of this table, it can be seen that, comparable values of D_w are obtained from the principal component score-based representation in sufficiently lower dimensions. For example, the D_w values obtained from 6 principal component scores are at least equal to the values obtained from 55 moments at noise levels $\sigma = 10, 20, 30$ and 40, and 136 moments at noise level $\sigma = 50$. From these observations it can be concluded that, significant improvement in representation can be achieved by applying PCA on Krawtchouk moments also.

It can be observed from the summary of recognition results presented in Table

Table 3.8: The maximum values of D_w computed from Krawtchouk moments and principal component scores of the Krawtchouk moments

Noise level (σ)	Krawtchouk Moment			Principal Component Score		
	Maximum D_w	Moment order	No. of moments	Maximum D_w	No. of PC scores (N_p)	
10	7.46	9	55	12.75	96	6
20	2.90	9	55	5.61	96	6
30	1.37	9	55	3.22	96	6
40	0.48	9	55	1.91	96	6
50	-0.03	15	136	1.14	96	6

3.6 that PCA-based representations using Krawtchouk moments also perform better over the moments (Krawtchouk)-based representation. For example, minimum number of Krawtchouk moments needed for 100% recognition accuracy at noise level $\sigma = 30$ is 6 (order 2), whereas a minimum of 4 PC scores of Krawtchouk moments are sufficient to get the same recognition accuracy at that noise level.

3.5 Linear Discriminant Analysis for Improved Classification

The PCA finds components that are useful for representing data, but these components may not be useful for discriminating data in different classes. On the other hand, the Linear Discriminant Analysis (LDA) finds directions that are useful for discrimination. It (LDA) is a well-known method for dimensionality reduction and classification. It transforms high-dimensional data onto a low-dimensional space where the data achieves maximum class separability [33], [46], [56], [89], [158]. The new features derived in LDA are linear combination of the original features where the coefficients are from the transformation matrix. The goal of LDA is to minimize the within-class distance and to maximize the between-class distance. In this section, we present a study on use of LDA on

moment vectors for improved classification of printed Assamese characters. The following subsections, present the theory of LDA and experimental results of LDA on moment vectors of printed Assamese characters.

3.5.1 Linear Discriminant Analysis

Given a data set consisting of N_s samples where each sample is a d-dimensional feature vector belonging to one of the n different classes. Let $\mathbf{X_{ri}}$ be a vector representing the ith sample belonging to the rth class where $r = 1, 2, \ldots, n$, $i = 1, 2, \ldots, n_r$ and n_r is the number of samples in the rth class. Classical LDA computes a linear transformation $\mathbf{G}$ which maps $\mathbf{X_{ri}}$ in the d-dimensional space to a vector $\mathbf{X_{ri}^L}$ in d^l-dimensional space: $\mathbf{X_{ri}^L} = \mathbf{G}^T\mathbf{X_{ri}}, (d^l < d)$. In discriminant analysis, three scatter matrices, called *within-class, between-class* and *total* scatter matrices, denoted by $\mathbf{S_w}, \mathbf{S_b}$ and $\mathbf{S_t}$ respectively, are defined as follows [46]:

$$\mathbf{S_w} = \frac{1}{N_s}\sum_{r=1}^{n}\sum_{i=1}^{n_r}(\mathbf{X_{ri}} - \bar{\mathbf{X}}_\mathbf{r})(\mathbf{X_{ri}} - \bar{\mathbf{X}}_\mathbf{r})^T \tag{3.47}$$

$$\mathbf{S_b} = \frac{1}{N_s}\sum_{r=1}^{n} n_r(\bar{\mathbf{X}}_\mathbf{r} - \bar{\mathbf{X}})(\bar{\mathbf{X}}_\mathbf{r} - \bar{\mathbf{X}})^T \tag{3.48}$$

$$\mathbf{S_t} = \frac{1}{N_s}\sum_{r=1}^{n}\sum_{i=1}^{n_r}(\mathbf{X_{ri}} - \bar{\mathbf{X}})(\mathbf{X_{ri}} - \bar{\mathbf{X}})^T \tag{3.49}$$

where $\bar{\mathbf{X}}_\mathbf{r}$ is the *centroid* of the rth class, and $\bar{\mathbf{X}}$ is the *global centroid* and are defined as:

$$\bar{\mathbf{X}}_\mathbf{r} = \frac{1}{n_r}\sum_{i=1}^{n_r}\mathbf{X_{ri}} \tag{3.50}$$

and

$$\bar{\mathbf{X}} = \frac{1}{N_s}\sum_{r=1}^{n}\sum_{i=1}^{n_r}\mathbf{X_{ri}} \tag{3.51}$$

It follows from the definition that

$$\mathbf{S_t} = \mathbf{S_b} + \mathbf{S_w}$$

The scatter matrices in the lower dimensional space resulting from the linear transformation $\mathbf{G}$ become

$$\mathbf{S_w^L} = \mathbf{G}^T\mathbf{S_w}\mathbf{G} \tag{3.52}$$

$$\mathbf{S_b^L} = \mathbf{G}^T\mathbf{S_b}\mathbf{G} \tag{3.53}$$

$$\mathbf{S_t^L} = \mathbf{G}^T\mathbf{S_t}\mathbf{G} \tag{3.54}$$

An optimal transformation $\mathbf{G}$ would in some sense maximizes the ratio of the between-class scatter to the within-class scatter. A simple scalar of scatter is the determinant of the scatter matrix. The product of the eigenvalues and equivalently, the product of the variances in the principal directions is the value of the determinant. Using this measure, the criterion function is obtained as [46], [33]

$$J(G) = \frac{|\mathbf{S_b}^L|}{|\mathbf{S_w}^L|} = \frac{|\mathbf{G}^T\mathbf{S_b}\mathbf{G}|}{|\mathbf{G}^T\mathbf{S_w}\mathbf{G}|} \tag{3.55}$$

The optimal $\mathbf{G}$ which maximizes the above criterion function consists of the top eigenvectors of $\mathbf{S_w}^{-1}\mathbf{S_b}$ corresponding to the nonzero eigenvalues, provided that the scatter matrix $\mathbf{S_w}$ is nonsingular [46]. Since $\mathbf{S_t} = \mathbf{S_b} + \mathbf{S_w}$, an equivalent solution consists of the top eigenvectors of $\mathbf{S_t}^{-1}\mathbf{S_b}$ corresponding to the nonzero eigenvalues [158]. A more general solution, in cases where $\mathbf{S_t}$ may be singular, consists of the top eigenvectors of $\mathbf{S_t}^{+}\mathbf{S_b}$ corresponding to the nonzero eigenvalues [158]. Here $\mathbf{S_t}^{+}$ denotes the pseudo-inverse of $\mathbf{S_t}$ [54]. When $\mathbf{S_t}$ is nonsingular, then $\mathbf{S_t}^{+} = \mathbf{S_t}^{-1}$.

3.5.2 Experimental Results

LDA is performed on the moment vectors consisting moments of order 0 to 30 of the dataset used in the experiments presented in the previous two sections. In these experiments also, we use a representation scheme similar to the scheme presented in Section 3.3, where the moment vectors are replaced by the LDA

features. The maximum values of D_w obtained from the LDA features along with those obtained from PCA are shown Table 3.9.

Table 3.9: The maximum values of D_w computed from PCA and LDA

Noise Level	Legendre Moment		Tchebichef Moment		Krawtchouk Moment	
(σ)	PCA	LDA	PCA	LDA	PCA	LDA
10	9.34	9.78	11.45	11.45	12.75	12.70
20	3.88	4.17	5.09	5.09	5.61	5.59
30	1.85	1.98	2.79	2.80	3.22	3.23
40	1.01	1.13	1.67	1.68	1.91	1.91
50	0.34	0.45	0.89	0.90	1.14	1.14

It may be observed, from this table that, the maximum values of D_w obtained from LDA features of Legendre moments are slightly higher than the maximum values of D_w obtained from PCA scores. On the other hand, the maximum values of D_w obtained from both PCA scores and LDA features are almost equal for Tchebichef and Krawtchouk moments. Therefore, it is found that, better discrimination between different classes is achieved using LDA on Legendre moments, but the discrimination is not improved by LDA over PCA for Tchebichef and Krawtchouk moments.

A set of experiments is also done to analyze the recognition performance of the representation schemes using LDA features on the same dataset used in Subsection 3.3.2 of 19600 noisy character images of 98 characters of Figure 3.1. The recognition is done using a minimum distance classifier. The moment vector of an input image of an unknown character is transformed by the LDA transformation matrix obtained during representation. Euclidean distances of this transformed vector from the centroids $\bar{\mathbf{X}}_\mathbf{r}$ of the clusters C_r $(r = 1, 2, \ldots, n)$ are computed and the input image is assigned to the cluster of nearest cluster centroid and is recognized to be the character corresponding to that cluster. It is observed from these experiments that the recognition performances of the LDA-based representation schemes are almost similar to those of PCA-based representation schemes. The numbers of LDA features, required to obtain 100%

recognition accuracy, are found to be same as the number of principal component scores for Tchebichef and Krawtchouk moments. On the other hand, for Legendre moments, 100% recognition accuracy is observed using comparatively less number of LDA features than the number of principal component scores.

3.6 Conclusion

In this chapter, a study on representation of printed Assamese characters using Legendre, Tchebichef and Krawtchouk moments is presented. It is observed that Krawtchouk moment based representation is the best among the three moments considered, followed by the Tchebichef and the Legendre moments. It can also be concluded that the principal component analysis can provide a considerable improvement in the representation of noisy characters. The chapter also presents a study on use of LDA on moment vectors for improved inter-class discrimination in representation of printed Assamese characters. It is observed that, in case of Legendre moments, performance of LDA is better than PCA. On the other hand, performances of both PCA and LDA are found to be same for Tchebichef and Krawtchouk moments. Though the experimentation is done only on Assamese characters, similar results can be expected in other scripts also. In this work only the representation aspects of noisy characters of a single font are considered in a simple framework. The aim of the work is to have a comparative performance analysis of the three moment methods in representation of noisy characters which may be useful for any moment-based OCR system. The actual optical character recognition scenario is more complex involving characters of different fonts and sizes. Some of these aspects are addressed in the following chapters.

Chapter 4

Representation of Printed Assamese Characters using Scale Invariants of Orthogonal Moments

4.1 Introduction

An optical character recognition system should be able to recognize characters irrespective of position and size of the character images. Therefore, it becomes necessary for an OCR, to use features which are invariant to position and size of the images. Orientation invariance is not as important as position and size for an OCR from practical point of view [93]. Position or translation invariance is also not very important if the segmentation of the characters are done in such a way that only the minimum rectangle containing the character images are considered. This can be easily done in binary document images using segmentation approach proposed in literature [25]. Therefore, scale or size invariance property is more important than position and orientation invariance for the features of an OCR. In this chapter, a study on representation of printed Assamese characters using scale invariants of Legendre and Tchebichef moments is presented.

M. K. Hu [58] first introduced moment invariants which are invariant with

respect to translation, scale and rotation. These invariants are widely used in image analysis and pattern recognition. These moment invariants are derived from geometric moments. However, kernel functions of geometric moments are not orthogonal. Therefore, they contain high degree of information redundancy and are sensitive to noise at high moment orders [149], [100]. Orthogonal moments are proved to be better than the geometric moments for image analysis because of the use of orthogonal kernel functions. Different invariant functions of these orthogonal moments are also proposed in the literatures by various researchers. Conventionally, there are two approaches used to obtain translation and scale invariants of orthogonal moments [68], [100], [28]:

1. Image normalization method (INM)
2. Indirect method (IDM)

The image normalization method first computes the normalization parameters. These normalization parameters are used to compute the moments of the standard image from the given image [68], [111], [4]. The normalization is done by using either geometric moments or complex moments. This method standardizes the images by setting its zeroth order moments to a predetermined value and then moments of the standard image is computed from its centroid by redefining the image coordinates.

In indirect method, the scale and translation invariants of moments are computed using the corresponding invariants of geometric moments. First, the orthogonal moment functions are expressed in terms of geometric moments and then the scale and translation invariants are derived by replacing the geometric moments with the corresponding invariants of geometric moments [121].

Both the above two methods derive translation and scale invariants of orthogonal moments indirectly using moments other than the corresponding orthogonal moments. The invariant functions are not derived using corresponding orthogonal kernels. These two methods also suffer from some drawbacks. The normalization parameters in INM may not always correspond to an exact transformation of the scaled image and therefore, the computed moments may differ from the true moments of the standard image. On the other hand, IDM is

computationally expensive [28].

Recently, a new set of translation and scale invariants of Legendre moments based on orthogonal Legendre polynomials have been introduced in [28]. Similarly, another set of translation and scale invariants of Tchebichef moments based on discrete orthogonal Tchebichef polynomials have been introduced in [160]. The translation invariants are derived from the respective central moments. The scale invariants are derived by algebraically eliminating the scale factors contained in the scaled Legendre or Tchebichef moments. The invariance properties of these scale and translation invariants are also established experimentally in these papers. However, to the best of author's knowledge, no extensive study on classification accuracy of these invariants has been reported. Some experimental results on classification accuracy of the two sets of invariants are reported in [160], but the experiments were done on a small subset of English alphabet only. Therefore, it is necessary to examine the representation and classification aspects of these invariants on a large data set. In this chapter, a study on representation aspects of the scale invariants of Legendre and Tchebichef moments is presented. The following two sections present the derivations of scale invariants of Legendre and Tchebichef moments proposed in [28] and [160] respectively. This is followed by experimental results and conclusions.

4.2 Scale Invariants(SI) of Legendre moments

The scaled Legendre moments of an image $f(x, y)$ of size $N \times N$, which is non-uniformly scaled with different factors, a and b, along x and y-axes respectively, can be defined as [28]:

$$L_{pq}^{\cdot} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{+1} \int_{-1}^{+1} P_p(ax) P_q(by) f(x, y) dx dy \qquad (a \neq b) \in (\mathbb{R} - \{0\}). \tag{4.1}$$

where $P_p(ax)$ and $P_q(by)$ are the scaled Legendre polynomials along x-axis and y-axis respectively, and $\mathbb{R}$ is the set of real numbers.

The scaled Legendre polynomials along x-axis can be expressed as a series of

decreasing powers of x as follows:

$$\begin{aligned} P_p(ax) &= B_{pp}a^p x^p + B_{p(p-2)}a^{p-2}x^{p-2} + B_{p(p-4)}a^{p-4}x^{p-4} \\ &+ B_{p(p-6)}a^{(p-6)}x^{(p-6)} + \cdots + B_{p\dot{k}}a^{\dot{k}}x^{\dot{k}} \end{aligned} \tag{4.2}$$

where $\dot{k} = 1$ if p is odd, and 0 otherwise.

The Legendre polynomial coefficient, B_{pk} is defined as

$$B_{pk} = (-1)^{(p-k)/2}\frac{1}{2^p}\frac{(p+k)!}{((p-k)/2)!((p+k)/2)!k!} \tag{4.3}$$

The relationship between the original and scaled Legendre polynomials can be obtained by rearranging Equation 4.2 as shown below:

$$\begin{aligned} \sum_{k=0}^{p}\eta_{pk}P_k(ax) &= a^p\sum_{k=0}^{p}\eta_{pk}P_k(x); \\ &(p-k) = even \end{aligned} \tag{4.4}$$

where η_{pk} is given by

$$\begin{aligned} \eta_{pk} &= \sum_{r=0}^{\dot{p}-2}\frac{-B_{(p-r)k}\eta_{p(p-r)}}{B_{kk}}; \\ \dot{p}-r &= even; \quad \dot{p} = (p-k) \geq 2. \end{aligned} \tag{4.5}$$

and $\eta_{pp} = 1$

Similarly, the scaled Legendre polynomials along y-direction can be obtained by using Equations 4.4 and 4.5 as follows:

$$\begin{aligned} \sum_{d=0}^{q}\eta_{qd}P_d(by) &= b^q\sum_{d=0}^{q}\eta_{qd}P_d(y); \\ &(q-d) = even \end{aligned} \tag{4.6}$$

The Equations 4.4 and 4.6 form the kernel of the scale invariants of Legendre moments. The invariants are expressed as a series of $(p+q)$th, $(p+q-2)$th, $(p+q-4)$th, etc. orders of original or scaled Legendre moments. The $(p+q)$th order invariants, denoted by ψ^L_{pq}, are defined as [28]:

$$\psi^L_{pq} = \sum_{k=0}^{p}\sum_{d=0}^{q}\left[\frac{\tau_{pq}}{\tau_{kd}}\eta_{pk}\eta_{qd}\dot{L}_{kd}\right] = a^{p+1}b^{q+1}\sum_{k=0}^{p}\sum_{d=0}^{q}\left[\frac{\tau_{pq}}{\tau_{qd}}\eta_{pk}\eta_{qd}L_{kd}\right] \tag{4.7}$$

where τ_{pq} is the normalizing constant of Equation 3.6 and is given by,

$$\tau_{pq} = \frac{(2p+1)(2q+1)}{(N-1)^2} \tag{4.8}$$

The scale factors, a and b present in Equation 4.7 can be canceled out using the following relations:

1. $\psi^L_{00} = ab\dot{L}_{00}$,
2. $\psi^L_{p0} = a^{p+1}b\sum_{k=0}^{p}\frac{\tau_{p0}}{\tau_{k0}}\eta_{pk}\dot{L}_{k0}$,
3. $\psi^L_{0q} = ab^{q+1}\sum_{d=0}^{q}\frac{\tau_{0q}}{\tau_{0d}}\eta_{qd}\dot{L}_{0d}$,

and subsequently the normalized scale invariants of Legendre moments, denoted by ω^L_{pq}, are derived as follows:

$$\omega^L_{pq} = \frac{\psi^L_{pq}\ (\psi^L_{00})^{\xi+1}}{\psi^L_{(p+\xi)0}\ \psi^L_{0(q+\xi)}} \tag{4.9}$$
$$p, q, \xi = 0, 1, 2, 3, \ldots.$$

The ω^L_{pq}s defined in Equation 4.9 are also called aspect ratio invariants. They can be applied to images with uniform as well as non-uniform scaling. For negative values of a and/or b, the above equation is used for inverted or reflected images.

The computation speed of these invariants is affected by the factorial functions in the coefficients η_{pq}. The following recurrence relations, proposed in [28], can be used to avoid the factorial functions and to get better computational performance:

$$B_{(p-2)k} = \frac{-(p-k)}{p+k-1}B_{pk}, \tag{4.10}$$

$$B_{p(k-2)} = \frac{-k(k-1)}{(p+k-1)(p-k+2)}B_{pk}. \tag{4.11}$$

Using these relations the entire set of polynomial coefficients B_{pk}, of fixed order p or index k, can be computed inside a single loop.

4.3 Scale Invariants (SI) of Tchebichef moments

The scaled Tchebichef moments of an image $f(x, y)$ of size $N \times N$, which is non-uniformly scaled with different factors, a and b, along x and y-axes respectively, can be defined as [160]:

$$\dot{T}_{pq} = ab \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \tilde{t_p}(ax)\tilde{t_q}(by)f(x, y) \qquad (a \neq b) \in (\mathbb{R} - \{0\}). \tag{4.12}$$

where $\tilde{t_p}(ax)$ and $\tilde{t_q}(by)$ are the scaled Tchebichef polynomials along x-axis and y-axis respectively, and $\mathbb{R}$ is the set of real numbers.

The discrete Tchebichef polynomial of Equation 3.15 can be written as:

$$\begin{aligned} t_p(x) &= \sum_{k=0}^{p} \frac{(p+k)!}{(p-k)!(k!)^2} < p - N >_{p-k} < x >_k \\ &= \sum_{k=0}^{p} A_{p,k} < x >_k \end{aligned} \tag{4.13}$$

where

$$A_{p,k} = \frac{(p+k)!}{(p-k)(k!)^2} < p - N >_{p-k} \tag{4.14}$$

and

$$< x >_k = (-1)^k(-x)_k = x(x-1)(x-2)\ldots(x-k+1) \qquad k \geq 1, \quad < x >_0 = 1 \tag{4.15}$$

From Equations 3.14 and 4.15, we get the following

$$\tilde{t_p}(x) = \sum_{k=0}^{p} \tilde{A}_{p,p-k} < x >_k \tag{4.16}$$

where

$$\tilde{A}_{p,p-k} = \frac{A_{p,k}}{\beta(p, N)} \tag{4.17}$$

As given in [29], $< x >_k$ of Equation 4.15 can be expanded as

$$< x >_k = \sum_{\imath=0}^{k} s(k,\imath)x^{\imath} \tag{4.18}$$

where $s(k,\imath)$ are the Stirling numbers of the first kind satisfying the following recurrence relations

$$\begin{aligned} s(k,\imath) = s(k-\imath,\imath-1) - (k-1)s(k-1,\imath) \\ k \geq 1, \quad \imath \geq 1 \end{aligned} \tag{4.19}$$

with

$$\begin{aligned} s(k,0) = s(0,\imath) = 0 \\ k \geq 1, \quad \imath \geq 1, \quad s(0,0) = 1 \end{aligned} \tag{4.20}$$

From Equations 4.19 and 4.16, we get

$$\begin{aligned} \tilde{t_p}(x) &= \sum_{k=0}^{p} \tilde{A}_{p,p-k} < x >_k \\ &= \sum_{k=0}^{p} \sum_{\imath=0}^{k} \tilde{A}_{p,p-k} s(k,\imath) x^{\imath} \\ &= \sum_{\imath=0}^{p} \sum_{k=0}^{n-\imath} \tilde{A}_{p,p-k} s(p-k,\imath) x^{\imath} \\ &= \sum_{\imath=0}^{p} C(p,\imath) x^{\imath} \end{aligned} \tag{4.21}$$

where

$$\begin{aligned} C(p,\imath) &= \sum_{k=0}^{p-\imath} \tilde{A}_{p,p-k} s(p-k,\imath) \\ &= \sum_{k=0}^{p-\imath} C_k(p,\imath) \end{aligned} \tag{4.22}$$

with

$$C_k(p,\imath) = \tilde{A}_{p,p-k} s(p-k,\imath) \tag{4.23}$$

Similarly, the scaled Tchebichef polynomials along x-axis can be expressed as follows

$$\tilde{t}_p(ax) = \sum_{i=0}^{p} C(p,i) a^i x^i \tag{4.24}$$

The relationship between $\tilde{t}_p(x)$ and $\tilde{t}_p(ax)$ can be derived from Equations 4.21 and 4.24 as follows

$$\sum_{k=0}^{p} \lambda_{p,k} \tilde{t}_k(ax) = a^p \sum_{k=0}^{p} \lambda_{p,k} \tilde{t}_k(x) \tag{4.25}$$

where

$$\lambda_{p,k} = \sum_{r=0}^{p-k-1} \frac{C(p-r,k)\lambda_{p,p-r}}{C(k,k)} \qquad 0 \leq k \leq p \tag{4.26}$$

and

$$\lambda_{p,p} = 1 \tag{4.27}$$

Using a similar approach, the scaled Tchebichef polynomials along the y-axis can be deduced as

$$\sum_{l=0}^{q} \lambda_{q,l} \tilde{t}_l(by) = b^q \sum_{l=0}^{q} \lambda_{q,l} \tilde{t}_l(y) \tag{4.28}$$

Then, the relationship between the original and scaled Tchebichef moments can be established as

$$\psi_{pq}^T = \sum_{k=0}^{p} \sum_{l-0}^{q} \lambda_{p,k} \lambda_{q,l} \dot{T}_{kl} = a^{p+1} b^{q+1} \sum_{k=0}^{p} \sum_{l=0}^{q} \lambda_{p,k} \lambda_{q,l} T_{kl} \tag{4.29}$$

Finally, the following scale invariants of Tchebichef moments can be obtained by eliminating the scale factors a and b as follows

$$\omega_{pq}^T = \frac{\psi_{pq}^T \, (\psi_{00}^T)^{\xi+1}}{\psi_{p+\xi,0}^T \, \psi_{0,q+\xi}^L} \qquad p,q = 0,1,2,\ldots, \qquad \xi = 1,2,3,\ldots \tag{4.30}$$

These scale invariants (SI) of Tchebichef moments can be used for images with uniform as well as non-uniform scaling. For negative values of a and/or b, the above equation is used for inverted or reflected images.

The following recurrence relations can be used to compute $C_k(p,i)$ in Equation 4.23 in order to reduce the computational complexity in calculating $C(p,i)$ defined in Equation 4.22.

$$C_k(p,i) = 0; \quad i \geq p-k \tag{4.31}$$

and

$$C_{k-1}(p,i) = \frac{k(2p-k+1)}{(p-N-k+1)(p-k+1)^2}\frac{s(p-k+1,i)}{s(p-k,i)}C_k(p,i) \qquad i \leq p-k; \quad 0 \leq k \leq p-1 \tag{4.32}$$

with

$$C_p(p,0) = \tilde{A}_{p,0} \tag{4.33}$$

4.4 Experimental Results

The first set of experiments is conducted to examine the invariance property of the scale invariants of Legendre and Tchebichef moments. An Assamese character of size 32×32 is expanded with a set of scaling factors a and b along x and y directions respectively. The values of scale invariants of Legendre moments, for some selected orders, are computed using Equation 4.9 as proposed in [28] and shown in Table 4.1. Similarly, the values of the scale invariants of Tchebichef moments, computed using Equation 4.30 as proposed in [160], are shown in Table 4.2. The deviation of the invariants, represented by the percentage spread from the corresponding means of the scale invariants ($sd/mean$) as proposed in [28], are also shown in this table. Here sd and $mean$ respectively denote the standard deviation and mean of the scale invariants. From the Tables 4.1 and 4.2, it can be seen that the values of the scale invariants of Legendre and Tchebichef moments remain almost unchanged under different uniform and non-uniform scaling.

Table 4.1: The scale invariants of Legendre moments for a non-uniformly scaled Assamese character

Image	Scale (a, b)	Scale Invariants							
		ω_{00}^L	ω_{20}^L	ω_{02}^L	ω_{11}^L	ω_{21}^L	ω_{12}^L	ω_{30}^L	ω_{03}^L
	original	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(1.5, 1)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(1, 1.5)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(1.5, 1.5)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0059
	(2, 1)	0.0131	0.0049	0.0077	0.0016	0.0032	0.0053	0.0038	0.0058
	(1, 2)	0.0131	0.0049	0.0076	0.0017	0.0032	0.0053	0.0038	0.0058
	(1.5, 2)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(2, 1.5)	0.0131	0.0049	0.0077	0.0016	0.0032	0.0053	0.0038	0.0059
	(2, 2)	0.0131	0.0049	0.0077	0.0016	0.0032	0.0053	0.0038	0.0058
	(3, 1)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(1, 3)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(3, 2)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
	(2, 3)	0.0131	0.0049	0.0077	0.0016	0.0032	0.0053	0.0038	0.0058
	(3, 3)	0.0131	0.0049	0.0077	0.0017	0.0032	0.0053	0.0038	0.0058
mean		0.0131	0.0049	0.00769	0.00167	0.0032	0.0053	0.0038	0.00581
sd		0	9.0E-19	2.7E-05	4.7E-05	9.0E-19	0	9.0E-19	3.6E-05
(*sd*/*mean*) %		0	1.8E-14	0.3474	2.8048	2.8E-14	0	2.4E-14	0.6246
Average (*sd*/*mean*) %		0.472100439							

Table 4.2: The scale invariants of Tchebichef moments for a non-uniformly scaled Assamese character

Image	Scale (a,b)	Scale Invariants							
		ω_{00}^{T}	ω_{20}^{T}	ω_{02}^{T}	ω_{11}^{T}	ω_{21}^{T}	ω_{12}^{T}	ω_{30}^{T}	ω_{03}^{T}
	original	0.2044	-0.0937	0.2145	0.0508	0.1513	-0.0317	-0.0741	-0.1192
	(1.5,1)	0.2024	-0.0929	0.2244	0.0499	0.1487	-0.0331	-0.0734	-0.1158
	(1,1.5)	0.2001	-0.0922	0.21	0.0508	0.149	-0.0317	-0.0737	-0.1167
	(1.5,1.5)	0.1982	-0.0914	0.2196	0.05	0.1463	-0.0331	-0.0731	-0.1134
	(2,1)	0.2014	-0.0925	0.2302	0.0495	0.1474	-0.0339	-0.0731	-0.114
	(1,2)	0.1979	-0.0915	0.2076	0.0509	0.1478	-0.0317	-0.0735	-0.1154
	(1.5,2)	0.196	-0.0907	0.2171	0.05	0.1452	-0.0331	-0.0729	-0.1122
	(2,1.5)	0.1972	-0.091	0.2253	0.0495	0.1451	-0.0339	-0.0728	-0.1116
	(2,2)	0.1951	-0.0903	0.2228	0.0496	0.144	-0.0339	-0.0725	-0.1104
	(3,1)	0.2006	-0.092	0.2367	0.0489	0.1459	-0.0347	-0.0727	-0.112
	(1,3)	0.196	-0.0907	0.2055	0.0509	0.1467	-0.0317	-0.0733	-0.1143
	(3,2)	0.1942	-0.0898	0.229	0.049	0.1425	-0.0347	-0.0722	-0.1085
	(2,3)	0.1931	-0.0895	0.2206	0.0496	0.1429	-0.0339	-0.0723	-0.1093
	(3,3)	0.1923	-0.0891	0.2267	0.049	0.1414	-0.0348	-0.0719	-0.1074
mean		0.00362	0.00134	0.00903	0.00072	0.00275	0.00118	0.00062	0.00333
sd		0.19778	-0.0912	0.22071	0.04989	0.14601	-0.0333	-0.0730	-0.1129
$(sd/mean)\%$		1.83058	1.46398	4.09150	1.44522	1.88648	3.53232	0.84418	2.94875
Average $(sd/mean)\%$		2.255377105							

Since the scale invariants are invariant to uniform and non-uniform scaling, it is expected that they can identify elongated, contracted and reflected non-symmetrical as well as symmetrical images. But, how well these descriptors can discriminate between different images is important for any pattern recognition problem. As reported in [160], these scale invariants of Legendre and Tchebichef moments can provide 100% recognition accuracy in a noise-free condition. The recognition accuracies for noisy images with 4% salt-and-pepper noise are reported to be 66.5% and 75.7% for the scale invariants of Legendre and Tchebichef moments respectively. For images with 1% salt-and-pepper noise, the recognition accuracies are 84.24% and 98.53% respectively for scale invariants of Legendre and Tchebichef moments. However, the experiments in [160] were done on a small subset English characters only. Further, the above results may not hold for another character set. Therefore it is worthwhile to examine the recognition or classification performance for the Assamese character set. The next two sets of experiments are conducted to examine the performance of these scale invariants in representation of printed Assamese characters. The first set of experiments are for noisy character and the second set for characters of different font sizes.

An experiment similar to the experiment conducted in Chapter 3 is done to see the clusters formed by the noisy images of different characters, represented by the scale invariants, are distinguishable or not. The database consists of 9800 noisy images used in Chapter 3 of 98 characters of Figure 3.1. The feature vectors, denoted by $\mathbf{X_{ri}}$ ($r = 1, 2, \ldots, 98$; $i = 1, 2, \ldots, 100$), consists of $d = (t+1)(t+2)/2$ scale invariants of order 0 to t of the ith sample of the rth character, where t is the maximum order of scale invariants used in the feature vectors. The scale invariants of Legendre and Tchebichef moments are computed using Equations 4.9 and 4.30. Then the centroids $\bar{\mathbf{X}}_\mathbf{r}$ and the radii R_r of each class are computed using Equations 3.42 and 3.43 respectively by replacing the moments with the corresponding scale invariants. Then the values $d_e(\bar{\mathbf{X}}_\mathbf{r}, \bar{\mathbf{X}}_\mathbf{s})$ and $(R_r + R_s)$ are computed for all pairs of clusters to find out the closest pair clusters and then the weighted distance between the closest pair of clusters D_w is computed using Equation 3.45 as explained in Section 3.3.

The experiment is repeated at different noise levels determined by the

standard deviation in the range of $5 - 50$ in steps of 5 and considering moment vectors of order $1, 2, \ldots, 30$.

From the results of these experiments, the maximum value of D_w, at noise level $\sigma = 5$, is found to be -31.29 using the scale invariants of Legendre moments. This indicates overlapping clusters formed by noisy images of different characters. The values of D_w decrease with increasing noise and the clusters at higher noise level are more indistinguishable. It has been observed, from results presented in Table 3.1, that the maximum value of D_w at $\sigma = 5$ using Legendre moments is 11.24 and the clusters were disjoint up to a noise level $\sigma = 35$. Therefore, it is found that, the performance of the scale invariants of Legendre moments are not as good as the Legendre moments in distinguishing noisy images of different characters and are not suitable for representation of noisy printed Assamese characters.

Similarly, the performance of the scale invariants of Tchebichef moments in distinguishing noisy printed Assamese characters are also found to be worse than the Tchebichef moments. The maximum value of D_w using scale invariants of Tchebichef moments is found to be -2.36 at noise level $\sigma = 5$. As expected, the values of D_w decrease with increasing noise level and the clusters are more indistinguishable at higher noise level. Form Table 3.2, it can be seen that the maximum value of D_w at $\sigma = 5$ using Tchebichef moments is 12.16 and the clusters were disjoint up to a noise level $\sigma = 40$. Therefore, similar to the scale invariants of Legendre moments, the scale invariants of Tchebichef moments are also not found to be suitable for representation of noisy printed Assamese characters since the clusters are found to be overlapping at all noise levels.

In order to examine the recognition performance of the scale-invariants, the dataset used in Subsection 3.3.2 of 19600 noisy images of 98 characters of Figure 3.1 is used. The recognition is done using a minimum distance classifier. The Euclidean distances of the vector containing the scale-invariants of moments of each character image of this dataset from the centroids $\bar{\mathbf{X}}_\mathbf{r}$ of the clusters C_r $(r = 1, 2, \ldots, n)$ are computed. The input image is then assigned to the cluster of nearest cluster centroid and is recognized to be the character corresponding to that cluster. From these experiments, the recognition performances of the

scale-invariants of both Legendre and Tchebichef moments are found to be very poor. Highest recognition accuracy using scale-invariants of Legendre moments at noise level $\sigma = 10$ is found to be 15.9%. The recognition accuracies reduces with increasing noise level. At noise levels $\sigma = 20, 30, 40$, and 50 are respectively $10.1\%, 8.4\%, 8.0\%$, and 7.3%. Similarly, the highest recognition accuracy using scale-invariants of Tchebichef moments at noise levels $\sigma = 10$, and 20 are 13.5% and 9.2% respectively. As expected, the recognition accuracy reduces when noise level increases.

Table 4.3: Number of Unambiguous Clusters (MR - Moments computed after Resizing the images to 32×32)

Order	Legendre moments			Tchebichef moments		
	Scale Invariants	Moments	MR	Scale Invariants	Moments	MR
3	12	44	38	30	30	39
4	9	45	41	37	39	41
5	10	47	47	38	43	47
6	11	46	45	40	47	44
7	10	46	44	39	46	44
8	10	44	46	37	46	45
9	11	44	44	30	46	43
10	9	44	44	26	44	44
11	10	45	44	24	45	44
12	7	45	46	21	44	46
13	9	42	43	17	44	43
14	8	43	40	12	43	40
15	8	42	38	11	38	38
16	11	40	38	4	40	39
17	9	39	34	5	41	34
18	9	38	34	4	37	33
19	10	32	32	3	38	31
20	10	31	27	3	34	27

The next set of experiments is conducted to see the performance of these scale invariants in representation of printed Assamese characters of various font-sizes.

For these experiments, a database of 364 character images 52 Assamese characters (first four rows of Figure 3.1) is considered. The dataset contains 7 images of varying font-sizes of each character. The character images are represented by vectors, each consisting of the scale invariants of order 0 to t computed from the images, in a d ($d = (t+1)(t+2)/2$) dimensional Euclidean space. Ideally, these vectors are clustered into 52 clusters using a hierarchical clustering algorithm. Each of these 52 clusters is assigned to characters of the alphabet depending on their constituent images. A cluster that contains images of a single character is assigned to that character. On the other hand, a cluster that contains images of more than one character is assigned to all those characters whose images are in that cluster. The clusters which are assigned to single characters are termed as *unambiguous clusters* and the other clusters which are assigned to multiple characters are called *ambiguous clusters.* The clustering and assigning them to characters of the alphabet are repeated for different values of t from 3 to 20 in steps on 1. The representations are compared on the basis of the number of ambiguous clusters. A representation, which results in smaller number of ambiguous clusters, is considered to be better than another with more number of ambiguous clusters. The number of unambiguous clusters in the representation with scale invariants of Legendre moments and Tchebichef moments along with the moments themselves are shown in table 4.3. The table also shows the number of unambiguous clusters obtained from representation with moments (Legendre and Tchebichef) of the character images resized to a predefined size (32×32). It can be observed from this table that the performances of the scale invariants in representation are very poor in comparison to the moments themselves. Among the two scale invariants, the Tchebichef moment based scale invariants are better than the scale invariants of Legendre moments. On the other hand, the performance of Legendre and Tchebichef moments are almost equal. It can also be observed that, for both Legendre and Tchebichef moments, there is no significant difference between the representation with moments from original images and resized (to a standard size) images so far as the number of unambiguous (or ambiguous) clusters is considered.

4.5 Conclusion

From the above discussion it may be concluded that the scale invariants (SI) of Legendre and Tchebichef moments are not suitable for representation of a large number of characters having minor variations. It is also observed that, in representation of printed Assamese characters of varying sizes, the moment-based representations result in less number of ambiguous clusters in comparison to scale invariant-based representations. The moments of different orders represent different spatial characteristics of the image intensity distribution. From Equations 4.7, 4.9, 4.29 and 4.30, it is seen that, these scale invariants are obtained as functions of corresponding moments of different orders and thereby they are unable to retain the individual characteristics of the moments. Therefore, though they are invariant to uniform and non-uniform scaling transformation, their performance in distinguishing similar characters images is very poor. It is also observed that, the effect of scaling of images on Legendre and Tchebichef moments are not very significant in terms of number of ambiguous clusters.

Chapter 5

Recognition of Printed Assamese Characters by Splitting of Characters using Orthogonal Moments

5.1 Introduction

An optical character recognition system should recognize characters of different font-types and font-sizes. It is observed, from the results of the experiments presented in the previous chapter, that the scale invariants of Legendre and Tchebichef are not good in distinguishing different printed Assamese characters. On the other hand, the performance of the moments is found to be better than the scale invariants in representation of printed Assamese character of different font-sizes. Therefore, it is necessary to develop new methods to recognize characters of multiple font-types and multiple font-sizes using moments.

In practical situations, we cannot expect exact match of the moment values, even for two images of the same character of same size, due to many factors like noise. Moreover, in pattern recognition problems, the feature spaces are divided into some regions during the training phase, and recognition of an input pattern is made according to region (rather than a point) to which it maps. Keeping

these points in mind, in this chapter we propose some methods for recognition of printed Assamese characters of multiple font-sizes and multiple font-types using orthogonal moments.

5.2 Limitations of Moment-based Methods

A pattern recognition system generally consists of two phases: training (learning) and recognition (classification). The training phase of a character recognition system identifies the appropriate features for representing the characters from a set of training images of each character of the alphabet and establishes the decision boundaries in the feature space which separate character images belonging to different characters. The moment vectors constitute the feature space in a moment-based character recognition system. In the recognition or classification phase, the trained system computes moments for an input image and determines the point in the feature space to which it maps. Depending upon the mapping of the point within a decision boundary the system identifies the image to correspond to one of the characters.

Theoretically, the recognition accuracy of moment-based methods should increase with the number of moments used to represent the characters. But, computational requirement also increases with the number of moments. Apart from computational requirement, the more serious problem is the approximation error contained in computation of higher order moments. The inaccuracy in the computation causes error in the mapping of the images leading to overlapping of the clusters for different characters. It is also evidenced from our experiments that the performance of the moment vectors (feature set) in representation of character images does not improve beyond a certain level by increasing the size of the moment vectors. It can be mentioned here that the performance of a feature set (representation space) is determined by how well patterns from different classes can be separated [61]. The approaches, presented in this chapter, attempt to overcome these limitations by splitting similar characters, which ordinary moment methods fail to distinguish, into multiple segments and extracting moments from some of these segments as distinguishing features.

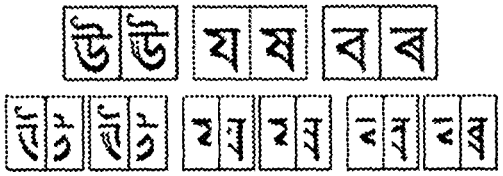


Figure 5.1: Three groups of similar Assamese characters (first row) and vertical splitting of them(second row)

5.3 Recognition by Splitting of Character Images

In character recognition, like many other pattern recognition problems, it has been observed that the distinguishing features, in many cases, are localized in small parts of a group of similar characters while the other parts are indistinguishable. For example, in recognition of characters of Assamese alphabet, the characters উ and ঊ, য and ষ etc. are some instances of such cases. In these cases, we can split the images, either vertically or horizontally or in both directions. The minute details of the images will be more prominent in one or more of the smaller parts. The Figure 5.1 shows three pairs of such characters of Assamese alphabet (first row) and their vertical splitting (second row). It can be observed from the second row this figure, that the right halves of the first two characters উ and ঊ are not distinguishable, whereas the left half of উ is clearly distinguishable from the left half of ঊ. On the other hand, the right halves of য and ষ are distinguishable but their left halves are not. Similarly, the right halves of ব and ৰ are also distinguishable but their left halves are not. The moment-based methods, generally, fail to distinguish such characters having minor, localized differences because the moments, in general, describe global features. If such characters are split into two or more parts, the localized distinguishing feature of them may become global in some of these small parts. Therefore, if the parts containing distinguishing feature(s) of the groups of similar characters are known

and moments are computed only from these parts, the moment-based methods will be able to distinguish these similar characters [127]. The Table 5.1 gives a list of groups of similar Assamese characters having localized distinguishing feature(s) in small parts. The halves containing distinguishing feature(s) of the the groups are indicated in the last column of this table. The list may vary depending on the moments and their order being used, because the distinguishing capabilities of different moments are not same and vary with order of moments. The list may also vary for datasets having characters of multiple font-types. Two moment-based approaches, based on these observations, are presented in next two sections.

Table 5.1: Groups of similar Assamese characters having minor, localized differences

Sl. No.	Similar characters	Parts containing distinguishing feature(s)
1	উ, ঊ	left
2	য, ষ	right
3	ব, ৰ, ৱ	right
4	হ, হ্	top
5	ছ, হ্	left or top
6	প, গ	right
7	ভ, ড	top
8	ঙ্ক, স্ক	top
9	ণ্ড, শু	left or top
10	ন্দ, ন্দ	left
11	ন্ধ, দ্ধ	left or top
12	ল, ন	left
13	গ, ণ	left
14	ক, ফ	left or top
15	ণ্ড, ণ্ড	left or top
16	ঠ, ষ্ঠ	left

5.4 Recognition of Characters of Single Font with Multiple Font-sizes (Split-character approach)

The success of a pattern recognition method depends on the distinctiveness of the extracted features. These distinguishing features, for example, are extracted by applying the filters in different directions in the Gabor-filter based approaches [19]. Our approach is based on splitting each ambiguous character into multiple segments and extracting moments from these segments as distinguishing features.

The proposed approach uses two different representation spaces (moment sets). In the first representation space, the points are the moment vectors computed from the character images. The points in the other representation space are also moment vectors but these moment vectors are computed from one half of each character image. Both training and recognitions are done in two stages. The training phase identifies the disjoint clusters in the first representation space and assigns each of them to one or more characters of the alphabet during the first stage. The second representation space is used in the second stage. This representation space is used for the characters whose corresponding clusters contain character images from other characters. The training and recognition procedures of the proposed approach are described below.

5.4.1 Training Phase (Learning)

Suppose the alphabet contains n characters. For each character k different character images are used to train the system.

5.4.1.1 First Stage

During the first stage of training phase, moments are computed from the character images of the training dataset. Each character image is represented by a vector, consisting of moments of order 0 to t computed from the image, in the first representation space. Ideally, these moment vectors are clustered into n clusters, denoted by C_i $(i = 1, 2, \ldots, n)$, using a hierarchical clustering algorithm. Each

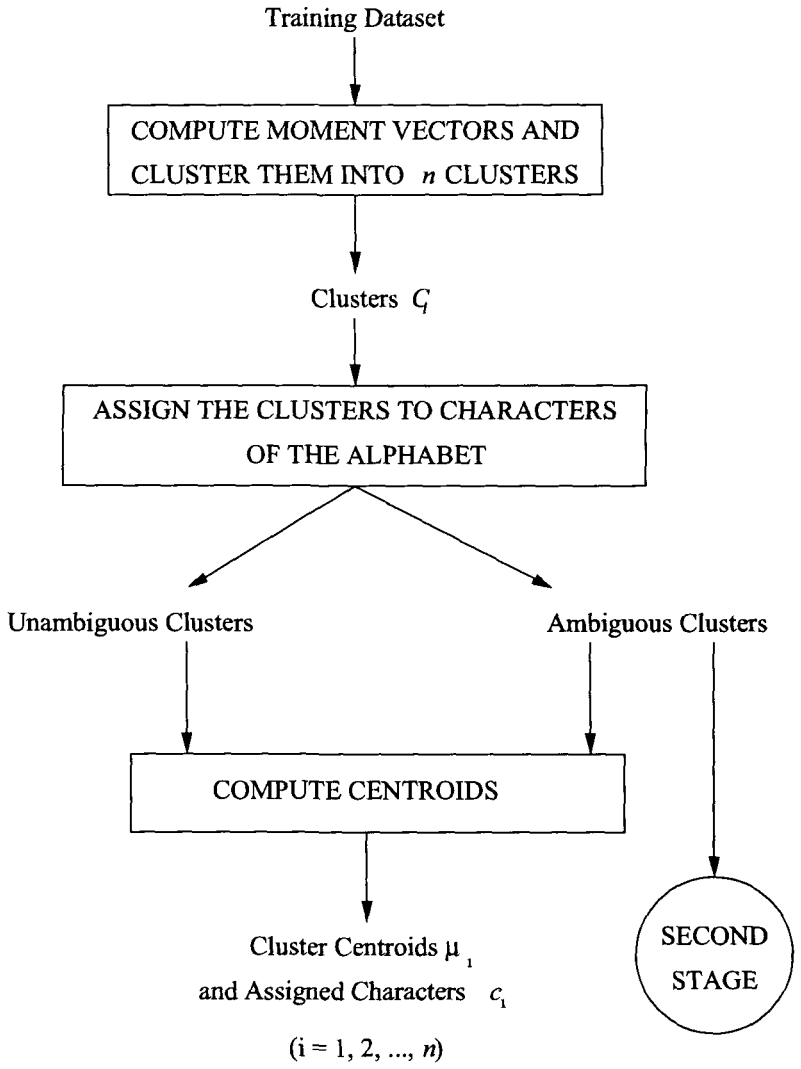


Figure 5.2: First stage of the training phase of the proposed split-character approach

of these n clusters are assigned to characters of the alphabet depending on their constituent images. A cluster that contains images of a single character is assigned to that character. On the other hand, a cluster that contains images of more than one character is assigned to all those characters whose images are in that cluster. The clusters which are assigned to single characters are termed as *unambiguous clusters* and the clusters which are assigned to multiple characters are called *ambiguous clusters*. The clustering and assigning them to characters of the alphabet are repeated incrementing t by 1 from an initial value till there

is improvement in clustering. An improved clustering means smaller number of ambiguous clusters. The moment order, beyond which there is no improvement in clustering, is termed as *optimal order of first stage (t_{opt})*. After that, the final assignment of the clusters is done and the mean vectors $\boldsymbol{\mu}_i$ $(i = 1, 2, \ldots, n)$ of the clusters are computed. These mean vectors are used as representatives of the respective clusters in the testing or recognition phase. The final assignment of the clusters to characters is represented by an array c_i $(i = 1, 2, \ldots, n)$. A non-zero positive value c_i of this array indicates that the cluster C_i is assigned to the c_ith character of the alphabet. On the other hand, an ambiguous cluster is indicated by a zero in this array. The first stage of the training phase (after deciding t_{opt}) is illustrated in Figure 5.2.

5.4.1.2 Second Stage

The second representation space is used in second stage and is required for the characters in the ambiguous clusters. Since moments represent global shapes of the images, therefore the characters in the ambiguous clusters have similar global shapes having minor local variations. Depending on the location of the minor variations, the images of these characters are split either vertically or horizontally into two equal parts. The parts which contain the distinguishing feature(s) are used in the second representation space. The location of the distinguishing feature is determined by visually observing the characters in the ambiguous clusters. For example, suppose an ambiguous cluster contains images of the two characters ঊ and ঙ whose left halves are different but right halves are similar. Therefore, the left halves of these two characters are used in the second representation space by splitting the characters vertically. After determining the appropriate halves for the characters in all the ambiguous clusters, moments (of order 0 to t') are computed from these partial images. The values of the array c_i for ambiguous clusters are modified to indicate the appropriate halves used in second stage. The values $-1, -2, -3$ and -4 are used to indicate left, right, top and bottom halves respectively. The appropriate halves to be used in the second stage of recognition phase can be determined from these values. After computing the moments, these images are represented as moment vectors in the second representation space.

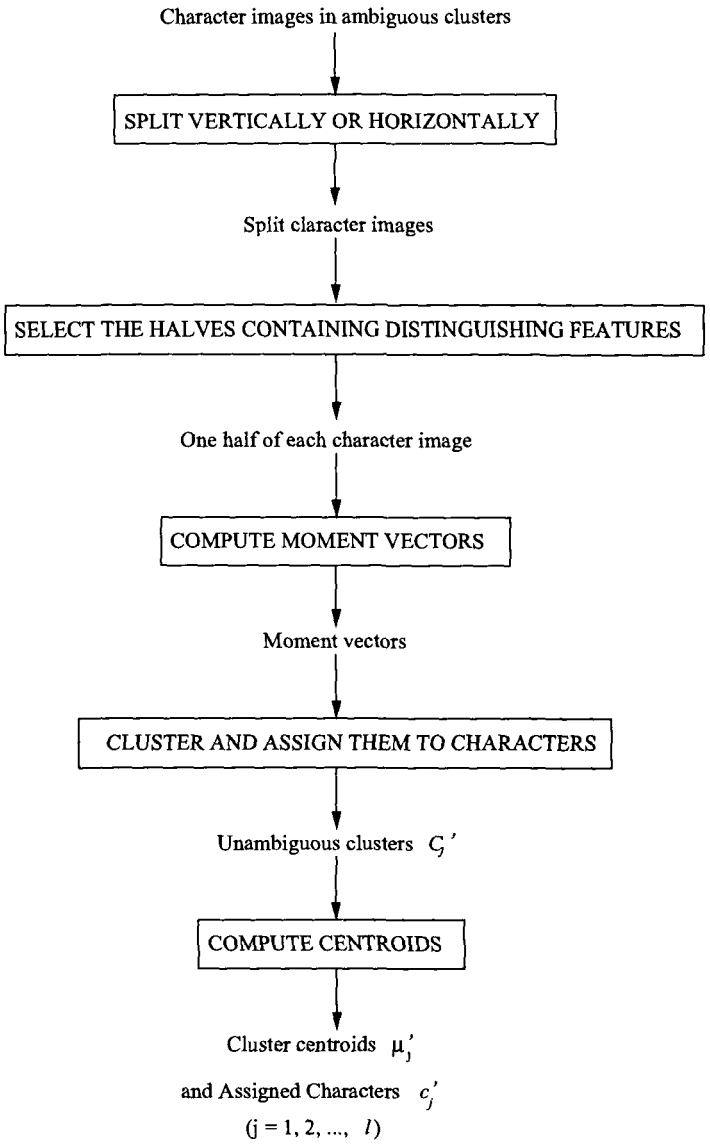


Figure 5.3: Second stage of the training phase of the proposed split-character approach

These moment vectors are clustered into l clusters, denoted by C_j' $(j = 1, 2, \ldots, l)$, where l is the total number of characters in all ambiguous clusters. The clusters are assigned to characters as done in the first representation space. The clustering and assigning of clusters to characters is repeated by increasing the moment order t' by 1 from an initial value till all clusters become unambiguous clusters or if there is no improvement of clustering after a certain number of repetitions. The

order, at which all the clusters become unambiguous or beyond which there is no improvement in clustering, is termed as *optimal order of second stage* (t'_{opt}). In the second representation space also, each cluster is represented by a mean vector $\boldsymbol{\mu}'_j$ $(j = 1, 2, \ldots, l)$. It is observed that, all clusters become unambiguous in the second representation space at very low moment orders. The assigned characters in the second stage are represented in an array c'_j, $(j = 1, 2, \ldots, l)$. The second stage of the training phase (after deciding t'_{opt}) is illustrated in Figure 5.3.

5.4.2 Recognition Phase

In this phase, we propose to use a minimum distance classifier. The two-stage recognition procedure is presented in Algorithm 5.1 and is described below.

5.4.2.1 First Stage

In the first stage of the recognition phase, the moment vector $\mathbf{x}$ (consisting of moments of order 0 to t_{opt}) of the input image I of an unknown character, is computed. After computing the moment vector $\mathbf{x}$, the Euclidean distances $\| \mathbf{x} - \boldsymbol{\mu}_i \|$ of $\mathbf{x}$ from each of the n mean vectors $\boldsymbol{\mu}_i$, $(i = 1, 2, \ldots, n)$ of the first representation space are computed and $\mathbf{x}$ is assigned to the cluster C_{min1} of the nearest mean vector $\boldsymbol{\mu}_{min1}$. If the assigned cluster C_{min1} is an unambiguous cluster, the character image is recognized as the character to which C_{min1} was assigned during the training phase. The assigned characters of the clusters of first stage are given in the array c_i, therefore, the input image I is recognized as the c_{min1}th character of the alphabet. On the other hand if C_{min1} is an ambiguous cluster, indicated by a negative value of c_{min1}, recognition is done in second stage.

5.4.2.2 Second Stage

The second stage of recognition is required for the input character images which are assigned to ambiguous clusters in the first stage. In this stage, the input character image is split either vertically or horizontally into two equal halves. The types of split (vertical or horizontal) and the appropriate halves to be used in second stage of recognition (for the ambiguous clusters) are already determined in

Algorithm 5.1 Recognition phase of proposed split-character approach

Input: $(\boldsymbol{\mu}_i, c_i)$ {(Cluster centroids, assigned characters) of first stage},
$(\boldsymbol{\mu}'_j, c'_j)$ {(Cluster centroids, assigned characters) of second stage},
I (input image); $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, l$

Output: r (input image is recognized as the rth character of the alphabet)

{*** FIRST STAGE ***}
$\mathbf{x}$:= moment vector of the input image I
$d_e(i) := \| \mathbf{x} - \boldsymbol{\mu}_i \|, \quad i = 1, 2, \ldots, n$
$min1 := \arg [\min_i(d_e(i))], \ \forall i = 1, 2, \ldots, n$ {* Find out the minimum distance cluster C_{min1} *}
if $(c_{min1} > 0)$ **then**
  $r := c_{min1}$ {* recognize I to be the c_{min1}th character of the alphabet *}
  return r
end if
{*** SECOND STAGE ***}
if $(c_{min1} = -1)$ **then**
  $I_a :=$ left half of I
else if $(c_{min1} = -2)$ **then**
  $I_a :=$ right half of I
else if $(c_{min1} = -3)$ **then**
  $I_a :=$ top half of I
else
  $I_a :=$ bottom half of I
end if
$\mathbf{x}'$:= moment vector of I_a
$d'_e(j) :=\| \mathbf{x}' - \boldsymbol{\mu}'_j \|, \quad j = 1, 2, \ldots, l$
$min2 := \arg [\min_j(d'_e(j))], \ \forall j = 1, 2, \ldots, l$ {* Find out the minimum distance cluster C'_{min2} *}
$r = c'_{min2}$. {* recognize I as the c'_{min2}th character of the alphabet *}
return r

the training phase and are given in c_i. After splitting the input character image I, a moment vector $\mathbf{x}'$ (consisting of moments of order 0 to t'_{opt}) is computed

from the appropriate half of the character image. Then, the Euclidean distances $\| \mathbf{x}' - \boldsymbol{\mu}'_j \|$ of $\mathbf{x}'$ from each of the l mean vectors $\boldsymbol{\mu}'_j$, $(j = 1, 2, \ldots, l)$ of the second representation space are computed, and $\mathbf{x}'$ is assigned to the cluster C'_{min2} of the nearest mean vector $\boldsymbol{\mu}'_{min2}$. Finally, the input character image is recognized as the c'_{min2}th character of the alphabet. The recognition phase is presented in Algorithm 5.1. In this algorithm the input c_is indicate the characters assigned to the clusters C_is, represented by μ_is of first stage, as described above. Similarly, input values c'_js indicate the characters assigned to the corresponding unambiguous clusters of second stage.

5.4.3 Experimental Results

5.4.3.1 Training Phase

A subset of Assamese alphabet, containing 79 most commonly used characters, is taken as the alphabet for the experiments. It consists of 11 vowels, 41 consonants and 27 conjunct (Yuktakshar) characters (Figure 5.4). The training dataset comprises of 395 character images of these 79 characters in 5 different font-sizes (14 points to 22 points in steps of 2 points). All the character images are taken of the same font. The size of the character images varies from 30×30 to 100×100. These 395 character images are represented in the first representation space and clustered into 79 clusters as described in Subsection 5.4.1. Assigning these clusters to different characters of the alphabet and labeling them as ambiguous or unambiguous clusters are also done as described in Subsection 5.4.1. This is repeated by varying the moment order t from 3 to 15. The number of ambiguous clusters against moment orders up to order 15 for Legendre, Tchebichef and Krawtchouk moments are presented in Figure 5.5.

Legendre Moments: It is observed from Figure 5.5 that, for Legendre moments, the clustering results improve initially for values of t from 3 to 6 and then remain unchanged for values of t from 6 to 15. The characters in ambiguous clusters for values of t from 3 to 8 are shown in Table 5.2. Clustering results for the values of t higher than 8 are not shown in this table because they are same as

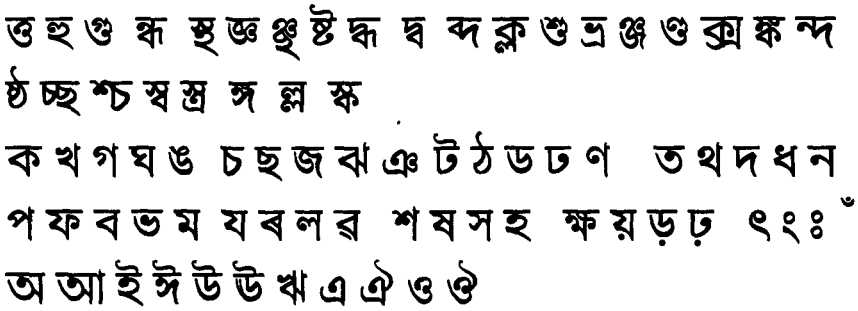


Figure 5.4: The Assamese characters used in the experiments; first two rows - conjunct characters, third and fourth rows - consonants and last row - vowels

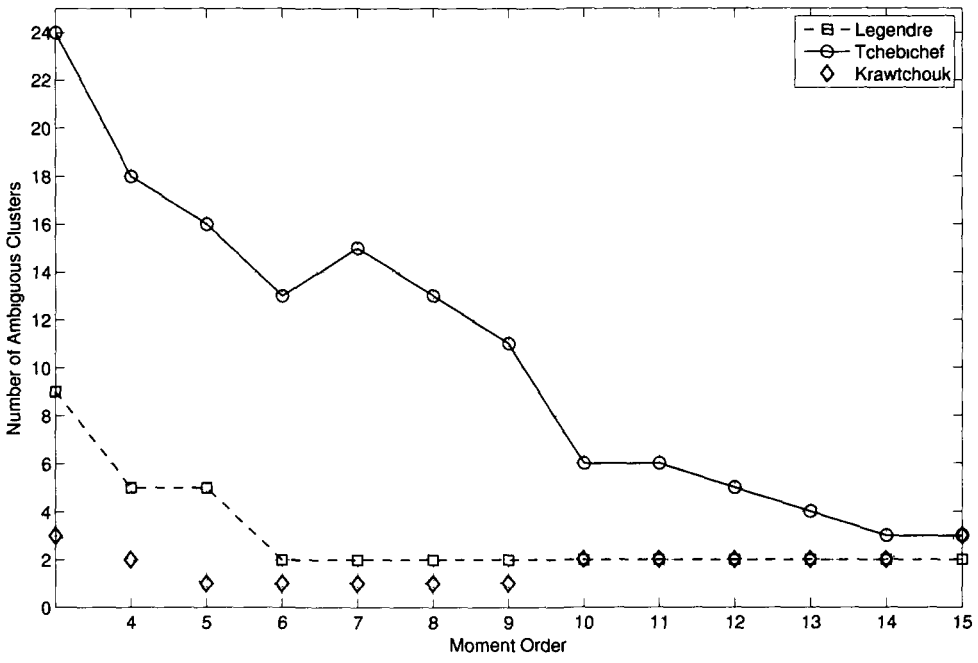


Figure 5.5: Number of Ambiguous Clusters using Legendre, Tchebichef and Krawtchouk Moments

the results at $t = 8$. More than one clusters may be formed by a set of characters and, in such cases, the number of ambiguous clusters is shown within brackets. Since there is no improvement in the clustering beyond $t = 6$, the value of t_{opt} is taken as 6 and the final assignment of the clusters is done at $t_{opt} = 6$ and the mean vectors $\boldsymbol{\mu}_i, i = 1, 2, \ldots, 79$ are computed for use as representatives of these clusters in the recognition phase.

Table 5.2: Characters in Ambiguous Clusters using Legendre Moments

Sl.	Order of moments					
No.	3	4	5	6	7	8
1	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ
2	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ
3	ল, ন	ল, ন	ল, ন			
4	ক্ষ, স্ক(2)		ক্ষ, স্ক			
5	ণ, শ	ণ, শ				
6		ক্ষ, স্ক	ক্ষ, স্ক			
7	ক, ফ					
8	প, শ					
9	ব, ঙ্গ					

There are two ambiguous clusters for $t \geq 6$ containing two characters in each cluster. Visual inspection reveals that the left halves of the two characters উ and ঊ in one of the ambiguous clusters are different whereas the right halves of these two characters are similar. On the other hand, the right halves of the two characters য and ষ in the other ambiguous cluster are different but the left halves are similar (Figure 5.1). Therefore, the left halves of the 10 character images of উ and ঊ and the right halves of the 10 character images of য and ষ are used in the second representation space. These 20 partial images of the four characters in two ambiguous clusters are represented in the second representation space using the moment vectors computed from them. Clustering and assigning the clusters to characters of the alphabet is done as described in the second stage of Subsection 5.4.1. It is found that, for $t = 4$, all the four clusters become unambiguous

clusters each assigned to one character. Therefore, the value of t'_{opt} is taken as 4 and the final assignment of the clusters is done at $t'_{opt} = 4$ and the mean vectors $\boldsymbol{\mu}'_j, j = 1, 2, \ldots, 4$ are computed to represent these four clusters in the second representation space.

Tchebichef Moments: It is observed from Figure 5.5 that, for Tchebichef moments, the clustering results improve for values of t from 3 up to 14 and then remain unchanged for values of $t = 15$. The characters in ambiguous clusters for values of t from 7 to 15 are shown in Table 5.3. Number of ambiguous clusters, in cases where more than one clusters are formed by a set of characters, are shown within brackets. The value of t'_{opt} is taken as 14 and the final assignment of the clusters is done at $t_{opt} = 14$ and mean vectors $\boldsymbol{\mu}_i, i = 1, 2, \ldots, 79$ are computed for use as representatives of these clusters in the recognition phase.

Table 5.3: Characters in Ambiguous Clusters using Tchebichef Moments

Sl.	Order of moments								
No.	7	8	9	10	11	12	13	14	15
1	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ	উ, ঊ
2	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ	য, ষ
3	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ	ব, ৰ
4	হ, হ	হ, হ	হ, হ	হ, হ	হ, হ	হ, হ	হ, হ		
5	ছ, হ	ছ, হ	ছ, হ						
6	প, গ	প, গ	প, গ						
7	ভ, ড	ভ, ড	ভ, ড						
8	ক, স্ক	ক, স্ক(2)	ক, স্ক(2)						
9	ণ, শ	ণ, শ							
10	ন্দ, ন্দ	ন্দ, ন্দ							
11	ন্ধ, দ্ধ								
12	ল, ন								
13			ব ৰ ব্ব	ব ৰ ব্ব					

There are three ambiguous clusters, each containing two characters, at $t = 14$.

Two of these clusters are same as those obtained with Legendre moments. It is seen from Figure 5.1 that the right halves of the two characters ব and ৰ in the third ambiguous cluster are different whereas the left halves of these two characters are similar. Therefore, the left halves of the 10 character images of উ and ঊ, and the right halves of the 20 character images of য, ষ, ব, and ৰ are used in the second representation space. These 30 partial images of the six characters in three ambiguous clusters are represented in the second representation space using the moment vectors computed from them. Clustering and assigning the clusters to characters of the alphabet is done as described in Subsection 5.4.1. It is found that for $t = 7$, all the six clusters become unambiguous each assigned to one character. Therefore the value of t'_{opt} is taken as 7 and the final assignment of the clusters is done for $t'_{opt} = 7$ and the mean vectors $\boldsymbol{\mu}'_j, j = 1, 2, \ldots, 6$ are computed to represent these four clusters in the second representation space.

Krawtchouk Moments: Krawtchouk moment-based representation results in the lowest number ambiguous clusters at all moment order. Like the Legendre and Tchebichef moments, Krawtchouk moment-based representation results in an ambiguous cluster containing উ and ঊ at all moment orders. Two other ambiguous clusters are found at $t = 3$. One of them consists of য and ষ and the other consists of ক্ষ, ক্ষ and ক্ষ. At $t = 4$, there are two ambiguous clusters. The second ambiguous cluster consists of ক্ষ and ৺. There is only one ambiguous cluster for $t = 5$ to 9, two for $t = 10$ to 14 and it becomes three at $t = 15$. Since there is no improvement in clustering performance beyond $t = 5$, the value of t_{opt} is taken as 5 and the final assignment of the clusters is done at $t_{opt} = 5$ and mean vectors $\boldsymbol{\mu}_i, i = 1, 2, \ldots, 79$ are computed for use as representatives of these clusters in the recognition phase. Considering the dissimilarity of the parts of উ and ঊ of the single ambiguous cluster at $t = 5$, the left halves of the 10 character images of these two characters are used in the second representation space. In the second representation space, it is found that the two clusters becomes unambiguous at $t = 1$ and therefore the value of t'_{opt} is taken as 1 and the final assignment of the clusters is done at $t'_{opt} = 1$ and the mean vectors $\boldsymbol{\mu}'_j, j = 1, 2, \ldots, 2$ are computed to represent the four clusters in the second representation space.

5.4.3.2 Recognition Phase

We conducted an experiment to test the proposed system. The test dataset consists of 1106 character images comprising all the 79 characters of Figure 5.4 in 14 different font-sizes varying from 14 point to 27 point. The size of the character images varies from 30×30 to 130×130.

Legendre Moment: The recognition accuracy with Legendre moments is found to be 99.64%. Recognition errors are found in the first stage of recognition. 4 character images out of 1106 are wrongly recognized in the first stage. All the 56 character images of উ, ঊ, য and ষ are correctly assigned to the corresponding ambiguous clusters in the first stage and then they are correctly recognized in the second stage. This shows 5.05% increase in recognition accuracy due to splitting of the characters of the ambiguous clusters.

Tchebichef Moment: The Tchebichef moment-based method also correctly recognizes 1102 character images out of 1106, resulting 99.64% recognition accuracy. The recognition errors occur in the first stage itself. All the 84 character images of the three ambiguous clusters are correctly recognized in the second stage indicating 7.59% increase in recognition accuracy due to splitting.

Krawtchouk Moment: The Krawtchouk moment-based method also results in 99.64% recognition accuracy. Like the other two moment methods the recognition errors occur in the first stage itself. All the 28 character images of the single ambiguous cluster are correctly recognized in the second stage indicating 2.53% increase in recognition accuracy due to splitting.

From the above results, it is found that the proposed split-character approach can recognize printed Assamese characters where ordinary moment-based methods fail and can correctly recognize characters having minute differences in their shapes. So far as the recognition accuracy is considered, all the three moments perform equally well. On the other hand, if the number of moments or equivalently the dimension of the representation space is considered, performance of Krawtchouk moments is the best followed by Legendre

and Tchebichef moments. Compared to the other two moments, Krawtchouk moments results in less number of ambiguous clusters in the first representation space at all moment orders. Therefore, dimensions of the representation spaces are minimum for Krawtchouk moments.

However, the performance of the split-character approach is not found to be satisfactory in recognition of characters of fonts other than those used in training. The recognition accuracies of this approach, when tested on a set of 395 characters images of a font other than training font, are found to be 66.3%, 77.2% and 68.6% using Legendre, Tchebichef and Krawtchouk moments respectively. It is also observed that, the split-character approach results in large number of ambiguous clusters in both first and second representation spaces even at very high moment orders when the training dataset includes character images of different font-types.

5.5 Recognition of Characters of Multiple Font-types and Multiple Font-sizes (split-character-with-LDA Approach)

A modified split-character approach, called *split-character-with-LDA*, is proposed to improve the performance in recognizing characters of multiple font-types. The proposed split-character-with-LDA approach performs Linear Discriminant Analysis (LDA) on the moment vectors to determine an appropriate projection of the moment vectors in order to improve the recognition accuracy. The approach also uses the concept of Euler Numbers to recognize characters in ambiguous clusters in some cases.

Similar to the split-character approach, the split-character-with-LDA approach is also based on splitting each ambiguous character into multiple segments and extracting moments from these segments as distinguishing features. The proposed approach also works in two stages. A common representation space is used for all character images in the first stage. The points in this space are the feature vectors obtained by performing Linear Discriminant Analysis (LDA) on the moment vectors computed from the character images. The training phase

identifies the disjoint clusters in this representation space, and assigns each of them to one or more characters of the alphabet. In the second stage, a separate representation space is used for each ambiguous cluster. The points in these representation spaces are the moment vectors computed from one half of the character images of the corresponding clusters. The approach also uses Euler Numbers of the character images of the ambiguous clusters to distinguish them in some cases. The training and recognition procedures of the proposed approach are described below.

In the next subsection, we briefly describe the concept of Euler Numbers and then we present the training phase, the recognition phase and the experimental results of proposed split-character-with-LDA approach in the remaining subsections.

5.5.1 Euler Number in Pattern Recognition

Euler Number (EN) or genus is one of the widely used topological descriptors used in many image analysis and visual inspection applications including document image analysis [30], [151], [90], [71]. For planar image, the Euler number is defined as the number of connected components minus the number of holes [57], [115], [50], [60]. In the proposed system, the Euler numbers are used to distinguish the characters in ambiguous clusters. For example, it is found in many representations that the images of the characters ব, ৰ and ৱ are mapped to a single cluster. The Euler numbers of these characters are 0, -1 and 1 respectively and therefore they can be easily differentiated by these numbers.

5.5.2 Training Phase (Learning)

Suppose the alphabet contains n characters. For each character, k character images of different font-types are used to train the system. The two stages of the training phase are illustrated in Figures 5.6 and 5.7 and also described below.

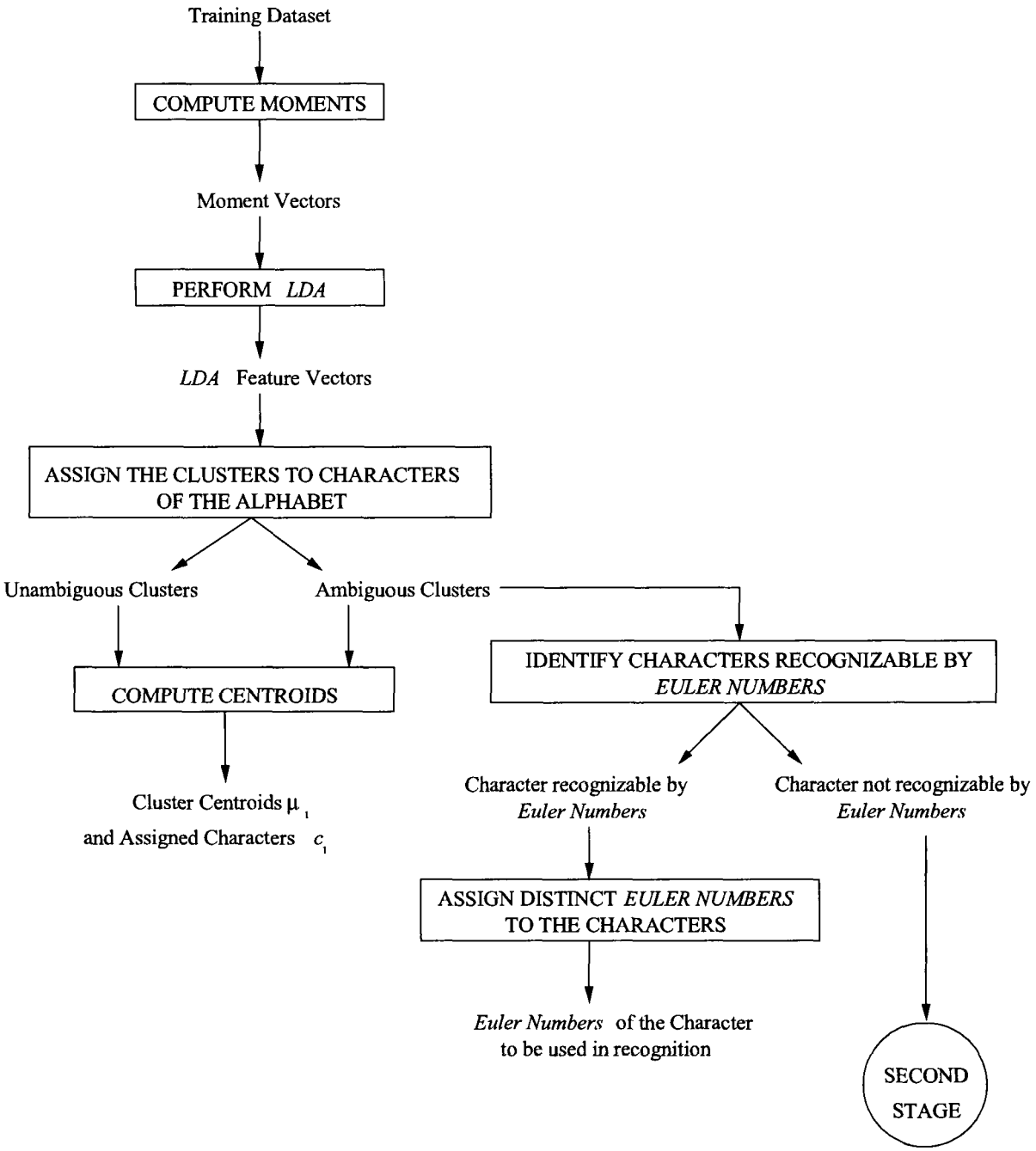


Figure 5.6: First stage of the training phase of the proposed split-character-with-LDA approach

5.5.2.1 First Stage

In the first stage of the training phase, the moments of order 0 to t are computed from each of the character images of the training dataset. New feature vectors are derived by performing LDA on these moment vectors and these feature vectors are used in the first representation space. Ideally, these feature vectors are clustered into n clusters, denoted by C_i, using a hierarchical clustering algorithm. Assigning these n clusters to the characters of the alphabet and identifying them as ambiguous or unambiguous clusters are done in the same way as done in split-character approach presented in Section 5.4. The clustering and assigning them to characters in the alphabet are repeated incrementing t by 1 from an initial value till there is improvement in clustering or up to a pre-specified order. As before, an improved clustering means smaller number of ambiguous clusters and an *optimal order of first stage (t_{opt})* is taken as the moment order beyond which there is no improvement in clustering. Sometimes, a lower moment order may be chosen as t_{opt}, if the number of ambiguous clusters obtained at this moment order is comparable with the minimum number of ambiguous clusters. Final assignment of the clusters is done at this optimal moment order (t_{opt}). After the final assignment, the mean vectors $\boldsymbol{\mu}_i$ and the array c_i, indicating the characters assigned to the clusters C_i ($i = 1, 2, \ldots, n$), are obtained in the same way as done in Section 5.4. The $\boldsymbol{\mu}_i$s are used as representatives of the clusters in the recognition phase. Finally, the characters in the ambiguous clusters are observed to see if they can be distinguished using Euler Numbers (EN). The ambiguous clusters, whose all constituent characters are distinguishable with EN, are now considered as unambiguous clusters. On the other hand, if all the constituent characters are not distinguishable by EN, the characters, which are clearly distinguishable by their EN from rest of the characters in an ambiguous cluster, are removed from the corresponding ambiguous cluster. In this case, the clusters with the remaining characters are still considered as ambiguous clusters.

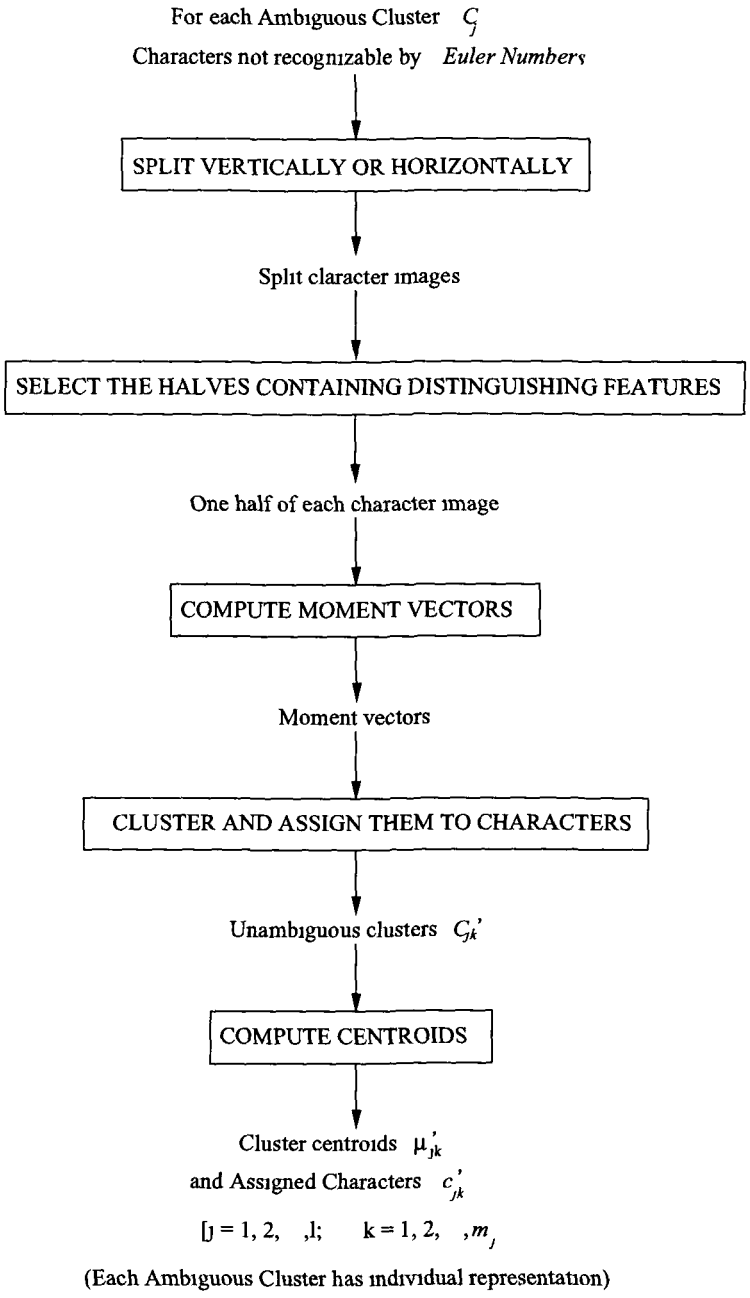


Figure 5.7: Second stage of the training phase of the proposed split-character-with-LDA approach

5.5.2.2 Second Stage

The second stage of the approach is required for characters in the ambiguous clusters which are not distinguishable by Euler Numbers. Depending on the

location of the distinguishing features, the images of these characters are split either vertically or horizontally into two equal parts and the parts which contain the distinguishing feature(s) are used in the representation spaces of second stage as done in Section 5.4. Similarly, the values of the array c_i for the ambiguous cluster are also modified to indicate the appropriate halves used in the second stage. The location and accordingly the appropriate split (vertical or horizontal) are determined by visually observing the characters. After determining the appropriate halves for all the characters in the ambiguous clusters which are not distinguishable with EN, moments (of order 0 to t') are computed from these partial images of the characters. These moment vectors are used to represent the character images of each ambiguous cluster in a Euclidean space. The character images are clustered into as many clusters as the number of characters assigned to the corresponding ambiguous clusters. For example, the moment vectors of character images of the ambiguous cluster $\mathcal{C}_j$ are clustered into m_j clusters denoted by $\mathcal{C}'_{jk}$ $(k = 1, 2, \ldots, m_j)$, where m_j is the number of characters assigned to $\mathcal{C}_j$. The clusters are assigned to characters as done in the first representation space. These clustering and assigning of clusters to characters are repeated by increasing the moment order t' till all clusters become unambiguous clusters or there is no improvement of clustering after a certain number of repetitions. In this approach, each ambiguous cluster has an *optimal order of second stage* denoted by $t'_{opt,j}$ ($j = 1, 2, \ldots, l$ and l is the number of ambiguous clusters), which are decided based on number of ambiguous clusters as done in first stage. The final assignment is done at these optimal orders. In the second representation space, the clusters are represented by the mean vectors $\boldsymbol{\mu}'_{jk}$ $(j = 1, 2, \ldots, l, k = 1, 2, \ldots, m_j)$, where l is the number of ambiguous clusters and m_j is the number of characters assigned to the jth ambiguous cluster $\mathcal{C}_j$. The characters assigned to the clusters are represented by an array c'_{jk}.

5.5.3 Recognition Phase

In this phase, we propose to use a minimum distance classifier. The two-stage recognition procedure is presented in Algorithm 5.2 and described below.

Algorithm 5.2 Recognition phase of proposed split-character-with-LDA approach

Input: $(\boldsymbol{\mu}_i, c_i)$ {(Cluster centroids, assigned characters) of first stage},
$(\boldsymbol{\mu}'_{jk}, c'_{jk})$ {(Cluster centroids,assigned characters) of second stage},
I {(input image)}; $\mathbf{G}$ {(LDA transformation matrix of training phase)}
$i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, l$; $k = 1, 2, \ldots, m_j$

Output: r (input image is recognized as the rth character of the alphabet)

{*** FIRST STAGE ***}
$\mathbf{x}$:= moment vector of the input image I
$\mathbf{ld} := \mathbf{G}^T\mathbf{x}$ {* Compute the LDA feature vector *}
$d_e(i) := \| \mathbf{ld} - \boldsymbol{\mu}_i \|, \quad i = 1, 2, \ldots n$
$min1 := \arg [\min_i(d_e(i))], \ \forall i = 1, 2, \ldots, n$ {* Find out the minimum distance cluster C_{min1} *}
if $(c_{min1} > 0)$ **then**
  $r := c_{min1}$ {* Recognize I as the c_{min1}th character of the alphabet *}
else if (C_{min1} contains characters recognizable by *Euler Number*) **then**
  $en :=$ *Euler Number* of I
  if (en is equal to one of the *Euler Numbers* assigned to C_{min1}) **then**
    $r :=$ the character assigned to en in C_{min1}
    return r
  end if
end if
{*** SECOND STAGE ***}
$I_a :=$ the half of I containing distinguishing feature {* Select the appropriate half based on the value of c_{min1} *}
$\mathbf{x}' :=$ moment vector of I_a
$d'_e(k) :=\| \mathbf{x}' - \boldsymbol{\mu}'_{min1,k} \|, \quad k = 1, 2, \ldots, m_j$
$min2 := \arg [\min_k(d'_e(k))], \ \forall k = 1, 2, \ldots, m_j$ {* Find out the minimum distance cluster $C'_{min1,min2}$ *}
$r = c'_{min1,min2}$ {* Recognize I as the $c'_{min1,min2}$th character of the alphabet *}
return r

5.5.3.1 First Stage

To recognize an image I of an unknown character, the moment vector $\mathbf{x}$ (consisting of moments of order 0 to t_{opt}) of I is computed and then a new feature vector $\mathbf{ld}$ is obtained by transforming $\mathbf{x}$ by the LDA transformation matrix $\mathbf{G}$ of training phase. After this, the Euclidean distances $\| \mathbf{ld} - \boldsymbol{\mu}_i \|$, of $\mathbf{ld}$ from each of the n mean vectors $\boldsymbol{\mu}_i$ $(i = 1, 2, \ldots, n)$ of the clusters of first stage of training phase, are computed and $\mathbf{ld}$ is assigned to the cluster C_{min1} of the nearest mean vector $\boldsymbol{\mu}_{min1}$. If the assigned cluster C_{min1} is an unambiguous cluster (i.e. $c_{min1} > 0$), the character image I is recognized as the c_{min1}th character of the alphabet. If the assigned cluster is an ambiguous cluster whose constituent characters are distinguishable by EN, then it is recognized by EN computed from the image. On the other hand, if it is an ambiguous cluster, recognition is done in second stage.

5.5.3.2 Second Stage

Let, the input character image I is assigned to an ambiguous cluster C_j in the first stage. The moment vector $\mathbf{x}'$ (consisting of moments of order 0 to $t'_{opt,j}$) is computed from the appropriate half of the character image. The appropriate halves for the ambiguous clusters are determined in the training or learning phase and can be obtained from the values of c_i. Then, the Euclidean distances $\| \mathbf{x}' - \boldsymbol{\mu}'_{jk} \|$ from $\mathbf{x}'$ to each of the m_j mean vectors $\boldsymbol{\mu}'_{jk}, k = 1, 2, \ldots, m_j$ of the second representation space are computed. The moment vector $\mathbf{x}'$ is assigned to the cluster $C'_{min1,min2}$ of the nearest mean vector $\boldsymbol{\mu}'_{min1,min2}$ and the input character I is recognized as the $c'_{min1,min2}$th character of the alphabet.

5.5.4 Experimental Results

5.5.4.1 Training Phase (Representation)

The alphabet for this set of experiments is same as the alphabet used in the Subsection 5.4.3 (Figure 5.4). The dataset for training phase comprises of 395 character images of the character of this alphabet in five different font-types.

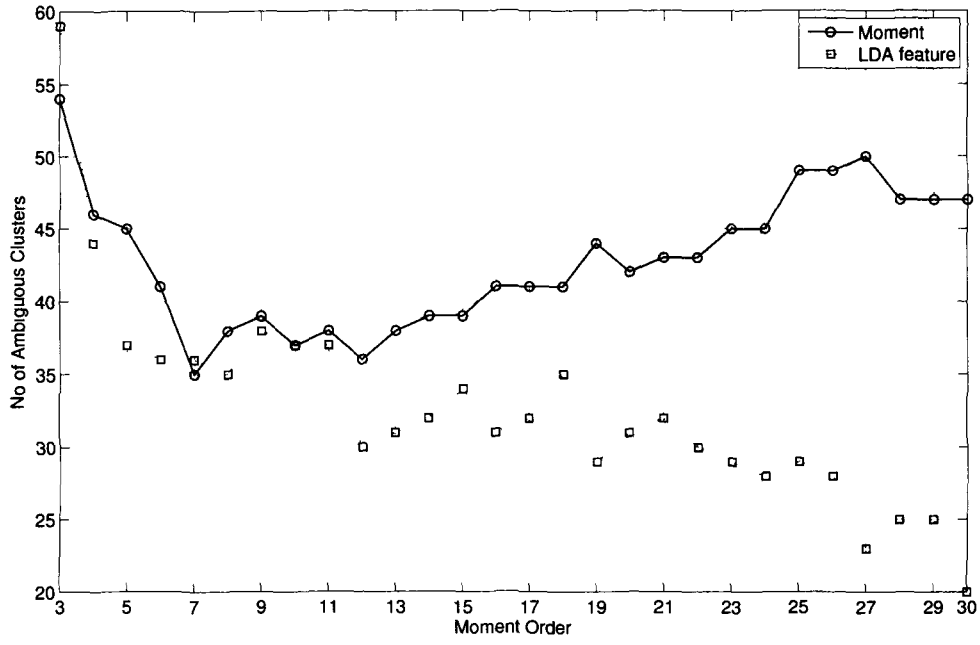


Figure 5.8: Number of ambiguous clusters using Legendre moments and LDA features of Legendre moments

These 395 character images are represented in the first representation space and clustered into 79 clusters as described in Subsection 5.5.2. Assigning these clusters to different characters of the alphabet and labeling as ambiguous and unambiguous clusters are also done as described in Subsection 5.5.2. This is repeated by varying the moment order t from 3 to 30. It has been observed that, the features obtained by LDA significantly improves the clustering performance in training phase. This is evidenced by the reduced number of ambiguous clusters obtained by the LDA features in comparison to the moment features. The results for Legendre, Tchebichef and Krawtchouk moments are presented in the following subsections.

Legendre Moments The clustering results for Legendre moments are presented in Figure 5.8. The plots in this figure show the number of ambiguous clusters against moment order. It is found that LDA features results in

Table 5.4: Characters in Ambiguous Clusters(AC) using LDA features of Legendre Moments up to order 12

Sl. No.	Characters in AC	No. of AC	Recognizable with EN	Characters to be split	Halves used in 2nd stage
1	উ, ঊ	1		উ, ঊ	Left
2	খ, থ	1		খ, থ	Top
3	ভ, ধ	1	ভ, ধ		
4	ঘ, য, স, ষ, য়	1	ষ, য়	ঘ, য, স	Left
5	ব, ঘ, য, ম, ষ, ৰ	1		ব, ঘ, য, ম, ষ, ৰ	Left
6	গ, জ, প	1	প	গ, জ	Left
7	গ, ণ, প	1	প	গ, ণ	Left
8	ল, ন	1		ল, ন	Left
9	ব, ৰ, ৱ	1	ব, ৰ, ৱ		
10	য, ম, ষ, য়	1	ষ, য়	য, ম	Left
11	চ, ঢ	2	চ, ঢ		
12	ৰ্ক, দ্ধ	1		ৰ্ক, দ্ধ	Top
13	স্থ, ষ্ণ	1		স্থ, ষ্ণ	Top
14	ঋ, শ্চ, স্ব, স্থ, অ, ঋ	1		ঋ, শ্চ, স্ব, স্থ, অ, ঋ	Left
15	ত্ত, হ্ন	1		ত্ত, হ্ন	Left
16	স্থ, ই	1	স্থ, ই		
17	ঞ্জ, হ্ম, ঔ	1	ঞ্জ, হ্ম, ঔ		
18	ষ্ঠ, ঠ	2	ষ্ঠ, ঠ		
19	স্ব, অ	1	স্ব, অ		
20	ণ্ড, ণ্ড	2		ণ্ড, ণ্ড	Left
21	ষ্ট, ই, ও, উ, ঠ	1	ষ্ট, ঠ	ই, ও, উ	Right
22	ৰ্ক, স্ক, ৰ্ক, ঞ্চ	1		ৰ্ক, স্ক, ৰ্ক, ঞ্চ	Top
23	ত্ত, হ্ন, ছ	1	ছ	ত্ত, হ্ন	Left
24	ত্ত, হ্ন, হ	1		ত্ত, হ্ন, হ	Left
25	ৰ্ক, ক, ফ	1	ৰ্ক, ক, ফ		
26	ন্দ, ন্দ	1	ন্দ, ন্দ		
27	ন্ত্র, ন্ন	1	ন্ত্র, ন্ন		

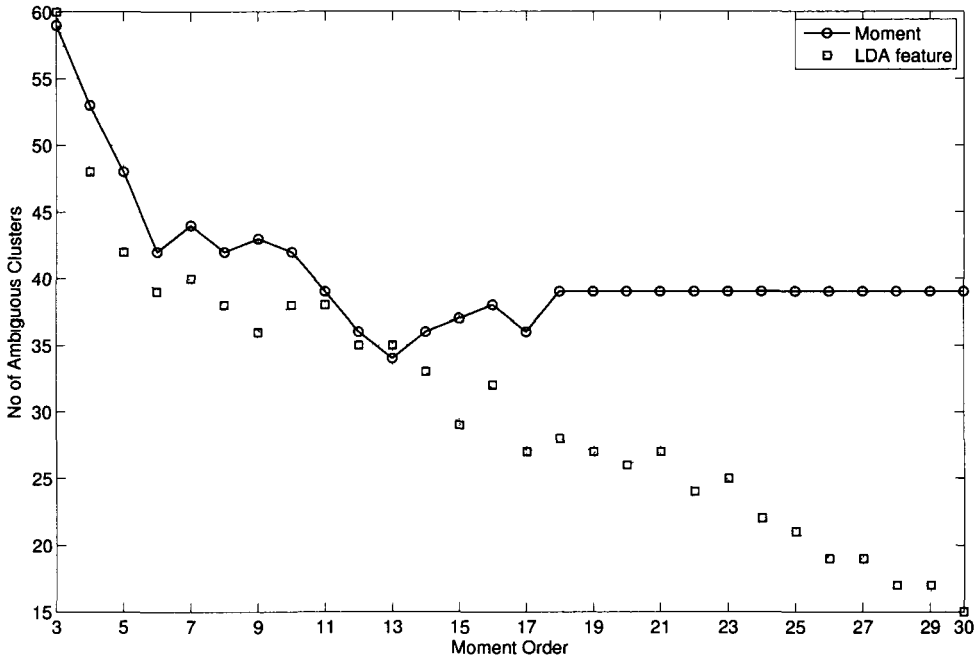


Figure 5.9: Number of ambiguous clusters using Tchebichef moments and LDA features of Tchebichef moments

significantly less number of ambiguous clusters than the moments. From the plot of LDA features, it is found that the number of ambiguous clusters is decreasing from order 3 to 8 and then it goes up at orders 9, 10 and 11. After that, it sharply decreases from 37 to 30 at order 12 and again it goes up after that. The number of ambiguous clusters shows a downward trend from order 18, but beyond order 19 the rate of decrease is very slow. Therefore, considering both number of ambiguous clusters and the size of the moment vectors, the optimal order of first stage (t_{opt}) is taken as 12 and the final assignment is done at order 12. The characters in ambiguous clusters, using LDA features of moments up to order 12, are shown in the Table 5.4. Each row of this table shows a set of characters that forms one or more ambiguous clusters. The set of characters that forms the cluster(s) is shown in the second column and the third column gives the number of such ambiguous clusters. The characters, which can be recognized by Euler Numbers, are shown in the fourth column and the characters to be split

in second stage are shown in the fifth column. The last column shows the halves, containing more prominent distinguishing features than the other parts, used in second stage.

Tchebichef Moments The clustering results for Tchebichef moments are presented in Figure 5.9. The plots in this figure show the number of ambiguous clusters against moment order. It is found that LDA features results in significantly less number of ambiguous clusters than the moments. From the plot of LDA features, it is seen that the number of ambiguous clusters has a steady decreasing trend from order 3 to 30 with minor fluctuations. It decreases fast from order 3 to 15 and then from order 15 to 30 the decreasing rate is slow. Though the minimum number of ambiguous clusters is observed at order 30, the optimal order of first stage (t_{opt}) is taken as 15 considering the size of moment vectors. The final assignment is done at order 15. The characters in ambiguous clusters, using LDA features of moments up to order 15, are shown in the Table 5.5. Similar to the Table 5.4, each row of this table shows a set of characters forming one or more ambiguous cluster in the first column and the number of such ambiguous clusters in the second column. The fourth column shows the character that can be recognized by EN and the fifth column shows the characters that are to be split in second stage. The last column shows the half of the characters, of the respective ambiguous clusters, used in second stage.

Krawtchouk Moments The clustering results for Krawtchouk moments are presented in Figure 5.10. The plots in this figure show the number of ambiguous clusters against moment order. It is found that LDA features results in significantly less number of ambiguous clusters than the moments. As in the case of Legendre and Tchebichef moments, the optimal order of first stage (t_{opt}) is decided considering both numbers of ambiguous clusters and size of the moment vectors. In this case, t_{opt} is taken as 14 in order to make the size of the moment vectors small, though lesser number of ambiguous clusters are observed at some higher orders. The characters in ambiguous clusters, using LDA features of moments up to order 14, are shown in the Table 5.6. Similar to the Tables 5.4 and 5.5, this table also shows the characters that can be recognized by Euler

Table 5.5: Characters in Ambiguous Clusters(AC) using LDA features of Tchebichef Moments up to order 15

Sl. No.	Characters in AC	No. of AC	Recognizable with EN	Characters to be split	Halves used in 2nd stage
1	ঊ, ঊ	1		ঊ, ঊ	Left
2	ঊ, ঊ, ই	1		ঊ, ঊ	Left
3	খ, থ	1		কহ, থ	Top
4	ভ, ড	1		ভ, ড	Top
5	য, ষ, য়	1	য, ষ, য়		
6	ব, ঘ, য, ষ, ৰ	1		ব, ঘ, য, ষ, ৰ	Left
7	গ, প	1	গ, প		
8	গ, ণ, প	1	প	গ, ণ	Left
9	ল, ন	1		ল, ন	Left
10	ব, ৰ, ৱ	1	ব, ৰ, ৱ		
11	চ, ঢ	2	চ, ঢ		
12	ক, ফ	1	ক, ফ		
13	ম, য়, ড়, ল্ল	1		ম, য়, ড়, ল্ল	Top
14	ত, ড, শ্চ	1		ত, ড, শ্চ	Top
15	জ, ঞ, জ্ঞ	1		জ, ঞ, জ্ঞ	Left
16	স্থ, স্ব	1		স্থ, স্ব	Right
17	ণ্ড, ণ্ড	2		ণ্ড, ণ্ড	Left
18	শু, ণ্ড	1	শু, ণ্ড		
19	জ, ঋ, ঙ্ক	1		জ, ঋ, ঙ্ক	Left
20	হ, ত্ত, হ্ন	1		হ, ত্ত, হ্ন	Top
21	ত্ত, হ্ন	1		ত্ত, হ্ন	Bottom
22	অ, অ	1		অ, অ	Left
23	ঙ্ক, স্ক, ক্ক	1	ঙ্ক	স্ক, ক্ক	Top
24	ঋ, ক্ক	1	ঋ, ক্ক		
25	ন্দ, ন্দ	1	ন্দ, ন্দ		
26	ঠ, ষ্ঠ	1	ঠ, ষ্ঠ		
27	হ্ন, ঙ	1	হ্ন, ঙ		

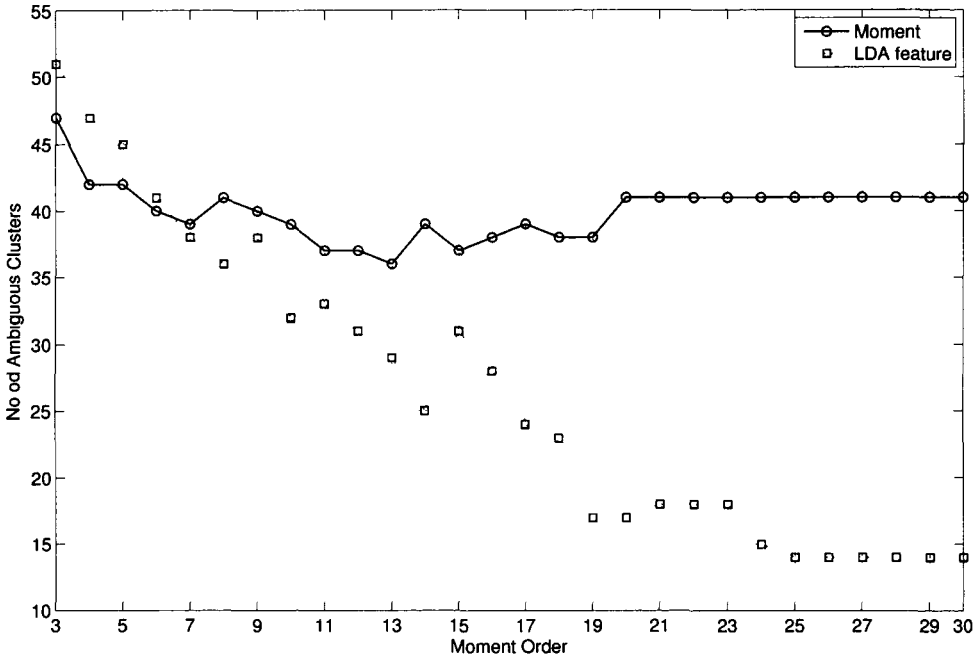


Figure 5.10: Number of ambiguous clusters using Krawtchouk moments and LDA features of Krawtchouk moments

Numbers, characters to be split in the second stage and the appropriate half to be used for recognition for characters in the ambiguous clusters in the respective columns.

5.5.5 Recognition Phase

The dataset for testing the proposed split-character-with-LDA approach consists of 1975 character images of the 79 printed Assamese characters given in Figure 5.4. The dataset contains 25 character images for each character of Figure 5.4 in 5 different font-types and in 5 different font-sizes. The summary of the recognition results using Legendre, Tchebichef and Krawtchouk moments is presented in Table 5.7. The first five rows of the table show the results of first stage, next two rows show the results of second stage and the combined results are shown in the last two rows. It is observed from this table that, the proposed

Table 5.6: Characters in Ambiguous Clusters(AC) using LDA features of Krawtchouk Moments up to order 14

Sl. No.	Characters in AC	No. of AC	Recognizable with EN	Characters to be split	Halves used in 2nd stage
1	ঊ, ঊ	2		ঊ, ঊ	Left
2	খ, থ	1		খ, থ	Top
3	ভ, ড	1		ভ, ড	Top
4	য, ষ, য়, ব, ৰ, ব়	1		য, ষ, য়, ব, ৰ, ব়	Left
5	ব, ঘ, য, ষ, ৰ	1		ব, ঘ, য, ষ, ৰ	Left
6	গ, ণ	1		গ, ণ	Left
7	ল, ন	1		ল, ন	Left
8	ব, ৰ, ব়	1	ব, ৰ, ব		
9	চ, ঢ	2	চ, ঢ		
10	ম, য়, ৱ	1		ম, য়, ৱ	Top
11	ছ, ত, হ	1	ছ	ত, হ	Left
12	ঠ, ষ্ঠ	1	ঠ, ষ্ঠ		
13	ঞ, ঙ, ত, জ্ঞ, ষ্ট	1		ঞ, ঙ, ত, জ্ঞ, ষ্ট	Left
14	হ্, শু, ও	1		হ্, শু, ও	Right
15	দ্ধ, ঞ্	1		দ্ধ, ঞ্	Left
16	ত, স্	2		ত, স্	Right
17	ত, হ্	1		ত, হ্	Left
18	হ্, স্থ, স্ব	1		হ্, স্থ, স্ব	Right
19	ণ্ড, ণ্ড	1		ণ্ড, ণ্ড	Left
20	ঙ্ক, স্ক	1		ঙ্ক, স্ক	Top
21	ঝ, ঋ, ষ্ট	1	ষ্ট	ঝ, ঋ	Left
22	ঔ, ট, ঋ, ন্দ	1	ঋ	ঔ, ট, ন্দ	Right

approach using Krawtchouk moments give the best results followed by Tchebichef and Legendre moments. Almost 97% recognition accuracy is achieved by the approach using Krwatchouk moments in this dataset. On the other hand, 95.19% and 96.56% recognition accuracy is achieved by the approach using Legendre

Table 5.7: Recognition results of split-character-with-LDA approach in a database of 1975 character images

		Legendre		Tchebichef		Krawtchouk	
		Number	%	Number	%	Number	%
First stage	Recognized using LDA feature of moments	925	46.84	1041	52.71	1153	58.38
	Recognized using EN	363	18.38	389	19.7	168	8.51
	Characters in ambiguous clusters	642	32.51	522	26.43	627	31.75
	Correct recognition	1288	65.22	1430	72.41	1321	66.89
	Wrong recognition	45	2.28	23	1.16	27	1.37
Second stage	Correct recognition	592	29.97	477	24.15	594	30.08
	Wrong recognition	50	2.53	45	2.28	33	1.67
Combined	Correct recognition	1885	95.19	1907	96.56	1915	96.96
	Wrong recognition	95	4.81	68	3.44	60	3.04

and Tchebichef moments respectively. Approximately, 30% character images are correctly recognized in second stage using both Legendre and Krawtchouk moments, while approximately 24% character images are correclty recognized using Tchebichef moments in second stage. On the other hand, the percentage of

characters recognized in first stage is more in case of Tchebichef moments than Legendre and Krawtchouk moments. It is also observed that a good number of characters are recognized by Euler numbers (18.38% using Legendre, 19.7% using Tchebichef and 8.51% using Krawtchouk moments).

5.6 Conclusion

In this chapter, two new approaches are presented to improve the recognition accuracy of moment-based methods. Both the methods utilize moment features from parts of the character images containing prominent distinguishing features to distinguish characters that are very similar in their shapes. Moment-based methods generally fail to distinguish characters having minor, localized variations in their shapes because the moments are global features. Therefore, improved recognition accuracy is obtained by taking the parts having more prominent distinguishing feature(s) and eliminating the parts having similar features in computing moments. Out of the two approaches, split-character approach uses moment vectors of the character images for representation and recognition of the characters and the split-character-with-LDA approach performs LDA and use the LDA features for representation and recognition of the character images in first stage and use the moment vectors in the second stage. Split-character-with-LDA approach also uses Euler numbers to recognize characters in some cases in first stage. Another aspect of the split-character-with-LDA approach, which is different from split-character approach, is the use of individual representation space for each ambiguous cluster in second stage. Clearly, the split-character approach is simpler than the split-character-with-LDA approach, but the former is not suitable for characters of multiple font-types and can recognize characters of multiple font-sizes of single font-types only. On the other hand, the split-character-with-LDA approach is suitable for both multiple font-sizes and multiple font-types. The proposed approaches can be extended to consider multiple splits of a character images and taking parts smaller than one half, in some cases, to improve representation and recognition accuracy. The improved approaches may also consider splitting of characters in non-rectangular and non-symmetrical parts. The study can also be extended to consider all characters of the script.

Chapter 6

Conclusions and Future Work

This chapter summarizes the major works done in this thesis and gives directions to further works. The thesis presents a study on some aspects of representation and recognition of printed Assamese characters using three orthogonal moments, namely, Legendre, Tchebichef and Krawtchouk moments. Experimental results on datasets of printed Assamese characters are presented. The datasets include all vowels and consonants and some commonly used conjunct (Yuktakshar) characters. The following two sections briefly describe the major achievements of this study and scopes for further work.

6.1 Major Achievements

6.1.1 Representation of Printed Assamese characters using Orthogonal Moments

A study on performance of Legendre, Krawtchouk and Tchebichef moments for representation of printed Assamese characters was taken up using a simple representation scheme. It was observed that, the performance of Krawtchouk moments is the best followed by Tchebichef and Legendre moments. It is also observed that, PCA and LDA can be used for better representation. For Legendre moments, LDA is found to give better class separability than PCA, whereas for Tchebichef and Krawtchouk moment, similar results are obtained using both

PCA and LDA.

6.1.2 Representation of Printed Assamese characters using Scale Invariants of Legendre and Tchebichef Moments

In this study, it was observed that the scale invariants of Legendre and Tchebichef moments are not good in distinguishing different printed Assamese characters from a large dataset in comparison to the respective moments. It is also observed that, effects of scaling on the moments in distinguishing different characters are not very significant and therefore the moments may be more suitable than the scale invariants for an OCR problem.

6.1.3 Development of new moment-based methods for recognition of printed Assamese characters

Two new moment-based methods for recognition of printed Assamese characters are proposed. The new methods can distinguish similar characters, having minor localized differences, which ordinary moment methods fail to distinguish. The observed recognition accuracies are in the range of 95% to 97%, which is comparable to many existing OCRs. However, the frequency of occurrence of all characters is same in the dataset for the experiments. Therefore, the accuracies may vary in a dataset, where the frequencies of occurrence of the characters in different documents are taken taken into account.

6.2 Further Work

There are many avenues for further work. Some of these avenues are outlined here.

Generally, LDA is better in class discrimination than PCA. However, there are some cases where PCA gives better class discrimination than LDA, particularly when the class samples do not adequately represent the classes. It was observed

in this study that, LDA gives better class discrimination for Legendre moments, whereas both LDA and PCA give almost similar results for Tchebichef and Krawtchouk moments. Further study may be taken up to find out the reasons for getting similar results using both PCA and LDA.

Representation of printed Assamese characters using scale invariants of Legendre and Tchebichef moments are studied here. Similar study may be taken up for scale invariants of Krawtchouk moments.

The new moment-based methods for printed Assamese characters split character images, in some cases, either vertically or horizontally and compute moments from one half having distinguishing feature(s). Further study may be taken up to consider multiple splitting, in some special cases, and compute moments from smaller parts. One may also consider splitting the characters in non rectangular or non symmetric parts.

In this study, vowel modifiers (also called matras) are not considered and may be included in future work.

Finally, similar studies may also be conducted using other moments and for other scripts.

Bibliography

[1] Y. S. Abu-Mostafa and D. Psaltis. Recognitive aspects of moment invariants. *IEEE Tuans. Pattern Analysis and Machine Intelligence*, 6(6):698–706, November 1984.

[2] T. Adamek, N. E. O'Connor, and A. F. Smeaton. Word matching using single-closed contours for indexing handwritten historical documents. *International Journal on Document Analysis and Recognition (IJDAR)*, 9(2):153–165, 2007.

[3] H. Al-Yousefi and S. S. Udpa. Recognition of Arabic characters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(8):853–857, 1992.

[4] M. Alghoniemy and A. H. Tewfik. Geometric distortion correction through image normalization. In *Proc. IEEE International Conference on Multimedia and Expo*, volume 3, pages 1291–1294, New York, USA, 2000.

[5] M. Altuwaijri and M. Bayoumi. Arabic text recognition using neural networks. In *Proc. Int. Symp. on Circuits and Systems (ISCAS-94)*, pages 415–418, 1994.

[6] A. Amin. Off-line Arabic character recognition: The state of the art. *Pattern Recognition*, 31(5):517–530, 1998.

[7] C. N. E. Anagnostopoulos, I. E. Anagnostopoulos, V. Loumos, and E. Kayafas. A license plate-recognition algorithm for intelligent transportation system applications. *IEEE Transaction on Intelligent Transportation Systems*, 7(3):377–392, September 2006.

[8] S. Antani and L. Agnihotri. Gujarati character recognition. In *Proc. Fifth International Conference on Document Analysis and Recognition(ICDAR '99)*, pages 418–421, Bangalore, India, September 1999.

[9] T. V. Ashwin and P. S. Sastry. A font and size independent OCR system for printed Kannada documents using support vector machines. *Sadhana*, 27:35–58, 2002.

[10] R. R. Bailey and M. Srinath. Orthogonal meoment features for use with parametric and non-parametric classifiers. *IEEE Pattern Analysis and Machine Intelligence*, 18:389–399, 1996.

[11] H. S. Baird. The state of the art of document image degradation modelling. In *Proc. of 4th IAPR International Workshop on Document Analysis Systems*, pages 1–16, 2000.

[12] R. Bajaj, L. Dey, and S. Chaudhury. Devnagari numeral recognition by combining decision of multiple connectionist classifier. *Sadhana*, 27:59–72, 2002.

[13] S. O. Belkasim, M. Shridhar, and M. Ahmadi. Shape-contour recognition using moment invariants. In *Proc. 10th International Conference on Pattern Recognition*, pages 649–651, 1990.

[14] S. O. Belkasim, M. Shridhar, and M. Ahmadi. Pattern recognition with moment invariants: A comparative study and new results. *Pattern Recognition*, 24:1117–1138, 1991.

[15] A. Bhardwaj, D. Jose, and V. Govindaraju. Script independent word spotting in multilingual documents. In *The 2nd International Workshop on 'Cross Lingual Information Access' Addressing the Information Need of Multilingual Societies*, pages 48–54, 2008.

[16] U. Bhattacharya, T. K. Das, A. Datta, S. K. Parui, and B. B. Chaudhuri. A hybrid scheme for handprinted numeral recognition based on a self-organizing network and MLP classifiers. *International Journal on Pattern Recognition and Artificial Intelligence*, 16:845–864, 2002.

[17] K. Bhattacharyya and K. K. Sarma. ANN-based innovative segmentation method for handwritten text in Assamese. *International Journal Computer Science Issues*, 5:9–16, October 2009.

[18] G. L. Cash and M. Hatamian. Optical character recognition by the method of moments. *Comput. Vision Graphics Image Process.*, 39(3):291–310, 1987.

[19] W. Chan and G. Coghill. Text analysis using local energy. *Pattern Recognition*, 34:2523–2532, 2001.

[20] M. Chandrasekaran. *Machine Recognition of The Tamil Script.* PhD thesis, University of Madras, India, 1982.

[21] S. L. Chang, L. S. Chen, Y. C. Chung, and S. W. Chen. Automatic license plate recognition. *IEEE Transaction on Intelligent Transportation Systems*, 5(1):42–53, March 2004.

[22] B. Chatterjee and A.K. Roy. On the classification of hand-printed Bengali numeral characters. In *Proc. Symposium on Microwaves andCommunication*, IIT Kharagpur, India, 1983.

[23] B. B. Chaudhuri and U. Pal. An OCR system for two indian language scripts: Bangla and Devnagari (hindi). In *Proc. Fourth International Conference on Document Analysis and Recognition*, pages 1011–1015, 1997.

[24] B. B. Chaudhuri and U. Pal. Skew angle detection of digitized Indian script documents. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 19:182–186, 1997.

[25] B. B. Chaudhuri and U. Pal. A complete printed Bangla OCR system. *Pattern Recognition*, 31:531–549, 1998.

[26] B. B. Chaudhuri, U. Pal, and M. Mitra. Automatic recognition of printed Oriya script. *Sadhana*, 27:23–34, 2002.

[27] P. Chinnuswamy and S. G. Krishnamoorty. Recognition of hand-printed Tamil characters. *Pattern Recognition*, 12:141–152, 1980.

[28] Chee-Way Chong, P. Raveendran, and R. Mukundan. Translation and scale invariants of Legendre moments. *Pattern Recognition*, 37:119–129, 2004.

[29] L. Comtet. *Advanced Combinatorics: The Art of Finite and Infinite Expressions.* D. Reidel Publishing Company, Dordrecht, Holland, 1974.

[30] D. B. Dhar and B. Chanda. Extraction and recognition of geographical features from paper maps. *International Journal of Document Analysis*, 8(4):232–245, 2006.

[31] R. Dhir. Feature extraction and classification for bilingual script (Gurmukhi and Roman). In *Proc. Third IASTED International Conference: Advances in Computer Science and Technology*, pages 336–341, April 2007.

[32] J. Dholakia, A. Negi, and S. Rama Mohan T. Zone identification in the printed Gujarati text. In *Proc. Eighth International Conference on Document Analysis and Recognition*, pages 272–276, 2005.

[33] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* John Wiley & Sons, Inc., 2nd edition, 2000.

[34] S. A. Dudani, K. J. Breeding, and R. B. McGhee. Aircraft identification by mmoment invariants. *IEEE Transactions on Computers*, 26:39–45, 1977.

[35] R. W. Duren. *An efficient second-order neural network architecture for orientation invariant character recognition.* PhD thesis, School of Engineering and Applied Science, Southern Methodist University, Dallas, TX, 1991.

[36] A. K. Dutta. A generalized formal approach for description and analysis for major Indian scripts. *Journal of Institute of Electronics and Telecom. Eng.*, 30:155–161, 1984.

[37] A. K. Dutta and S. Chaudhuri. Bengali alpha-numeric character recognition using curvature features. *Pattern Recognition*, 26:1757–1770, 1993.

[38] L. Eikvil. OCR - optical character recognition. Technical Report 876, Norsk Regnesentral: Norwegian Computing Center, January 1993.

[39] S. El-Dabi, R. Ramsis, and A. Kamel. Arabic charcter recognition system: statistical approach for recognizing cursive typewritten text. *Pattern Recognition*, 23(5):485–495, 1990.

[40] F. El-Khaly and M. A. Sid-Ahmed. Machine recognition of otically captured machine printed Arabic text. *Pattern Recognition*, 23(11):1207–1214, 1990.

[41] T. S. EL-Sheikh and R. M. Guindi. Computer recognition of Arabic cursive scripts. *Pattern Recognition*, 21(4):293–302, 1988.

[42] F. Feng and T. Pavlidis. Decomposition of polygons into simpler components: feature generation for syntactic pattern recognition. *IEEE Trans. Computer*, 24:636–650, 1975.

[43] J. Flusser. Moment invariants in image analysis. In *Proc. World Academy of Science, Engineering and Technology*, volume II, pages 196–201, 2006.

[44] J. Flusser and T. Suk. Affine moment invariants: A new tool for character recognition. *Pattern Recognition Letters*, 15:433–436, 1994.

[45] H. Fujisawa. Forty years of research in character recognition and document recognition - an industrial perspective. *Pattern Recognition*, 41(8), 2008.

[46] K. Fukunaga. *Introduction to Statistical Pattern Classification.* Academic Press, USA, 1990.

[47] U. Garain, B. B. Chaudhuri, and T. T. Pal. Online handwritten Indian script recognition: A human motor function based framework. In *Proc. 16th International Conference on Pattern Recognition*, volume 3, pages 164–167, 2002.

[48] H. Genchi, K. Mori, , S. Watanabe, and S. Katsuragi. Recognition of hand-printed numerical characters for automatic letter sorting. *Proceedings of IEEE*, 56:1291–1301, August 1968.

[49] R. K. Gerlach. Wide-tolerance optical character recognition for existing printing mechanisms. In G. L. Fischer at al., editor, *Optical Character Recognition*, pages 93–114. McGregor & Wemer, 1962.

[50] R. C. Gonzalez and R. E. Woods. *Digital Image Processing.* Addision-Wesley Publishing Company, 1993.

[51] L. O. Gorman and R. Kasturi. *Document Image Analysis.* IEEE Computer Society Press, Los Alamitos, CA, 1995.

[52] A. Goshtasby. Template matching in rotated images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 7:338–344, 1985.

[53] E. C. Greanias. Some important facotrs in the practical utilization of optical character readers. In G. L. Fischer at al., editor, *Optical Character Recognition*, pages 129–146. McGregor & Wemer, 1962.

[54] G. H. Gulab and C. F. Van Loan. *Matrix Computations.* The John Hopkins University Press, USA, 1996.

[55] P. W. Handel. Statistical machine. U.S. patent 1915993, June 1933.

[56] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning : Datamining, inference and Prediction.* Springer, 2001.

[57] B. P. K. Horn. *Robot Vision.* McGraw Hill, New York, 1986.

[58] M. K. Hu. Visual pattern recognition by moment invariants. *IRE Trans. on Information Theory*, IT-8:179–187, Feb 1962.

[59] H. Ishida, T. Takahashi, I. Ide, Y. Mekada, and H. Murase. Recognition of camera-captured low-quality characters using motion blurinformation. *Pattern Recognition*, 41(7):2253–2262, 2008.

[60] A. K. Jain. *Fundamentals of Digital Image Processing.* Prentice-Hall of India Pvt. Ltd., 1995.

[61] A. K. Jain, R. P. W. Duin, and J. Mao. Statistical pattern recognition: A review. *IEEE PAMI*, 22(1):5–37, Jan 2000.

[62] C. V. Jawahar, M. N. S. S. K. Pavan Kumar, and S. S. Ravi Kiran. A bilingual OCR for Hindi-Telugu documents. Technical report trcvit-22, IIIT, Hyderabad, India, 2002.

[63] J. Jiaoa, Q. Ye, and Q. Huanga. A configurable method for multi-style license plate recognition. *Pattern Recognition*, 42(3):358–369, March 2009.

[64] C. C. Heasly Jr and G. L. Fischer Jr. Some elements of optical scanning. In G. L. Fischer at al., editor, *Optical Character Recognition*, pages 15–26. McGregor & Wemer, 1962.

[65] S. Kahan, T. Pavlidis, and H. S. Baird. On the recognition of printed character of any font andsize. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9:274–288, 1987.

[66] C. Kan and M. D. Srinath. Invariant character recognition with Zernike and orthogonal Fourier-Mellin moments. *Pattern Recognition*, 35(1):143–154, January 2002.

[67] A. Khotanzad and Y. H. Hong. Rotation invariant pattern recognition using zernike moments. In *Proc. 9th ICPR*, pages 14–17, Rome, Italy, November 1988.

[68] A. Khotanzad and Y. H. Hong. Invariant image recognition by Zernike moments. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(5):489–497, 1990.

[69] A. Khotanzad and J.H. Lu. Classification of invariant image representations using a neural network. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 38(6):1028–1038, June 1990.

[70] T. Konidaris, B. Gatos, K. Ntzios, I. Pratikakis, S. Theodoridis, and S. J. Perantonis. Keyword - guided word spotting in historical printed documents using synthetic data and user feedback. *International Journal on Document Analysis and Recognition (IJDAR)*, 9(2):167–177, 2007.

[71] K. Kpalma and J. Ronsin. An overview of advances of pattern recognition systems in computer vision. In Goro Obinata and Asish Dutta, editors, *Vision Systems:Segmentation and Pattern Recognition*, pages 169–194. I-Tech, Vienna, 2007.

[72] S. Kumar and C. Singh. A study of Zernike moments and its use in Devanagari handwritten character recognition. In *Proc. International Conference on Cognition and Recognition*, pages 514–520, 2005.

[73] R. S. Kunte and R. D. Sudhaker Samuel. A simple and efficient optical character recognition system for basic symbols in printed Kannada text. *Sadhana*, 32(5):521–533, October 2007.

[74] R. C. Kurzweil. *Kurzweil Reading Machine for the Blind (User Manual).* Kurzweil Computers Products, Cambridge, M.A., 1990.

[75] V. Lavrenko, T. M. Rath, and R. Manmatha. Holistic word recognition for handwritten historical documents. In *Proc. of the First International Workshop on Document Image Analysis for Libraries (DIAL'04)*, pages 278–287, 2004.

[76] G. S. Lehal and R. Dhir. A range free skew detection technique for digitized Gurmukhi script documents. In *Proc. Fifth International Conference of Document Analysis and Recognition*, pages 147–152, 1999.

[77] G. S. Lehal and C. Singh. Feature extraction and classification for OCR of Gurmukhi script. *Vivek*, 12:2–12, 1999.

[78] G. S. Lehal and C. Singh. A Gurmukhi script recognition system. In *Proc. of the 15th International Conference on Pattern Recognition*, volume 2, pages 557–560, 2000.

[79] G. S. Lehal and C. Singh. Technique for segmentation of Gurmukhi text. In W. Skarbek, editor, *Computer Analysis of Images and Patterns, Lecture Notes in Computer Science*, volume 2124, pages 191–200. Springer, Germany, 2001.

[80] G. S. Lehal and C. Singh. Text segmentation of machine printed Gurmukhi script. In P. B. Kantor, D. P. Lopresti, and Jiangying Zhou, editors, *Document Recognition and Retrieval VIII, Proceedings SPIE*, volume 4307, pages 223–231, USA, 2001.

[81] G. S. Lehal and C. Singh. A post processor for Gurmukhi OCR. *Sadhana*, 27:99–111, 2002.

[82] G. S. Lehal, C. Singh, and R. Lehal. Shape based post processor for Gurmukhi OCR. In *Proc. Sixth International Conference on Document Analysis and Recognition*, pages 1105–1109, Seattle, 2001.

[83] S. X. Liao. *Image Analysis by Moments.* PhD thesis, University of Monitoba, Canada, 1993.

[84] S. X. Liao, Q. Lu A. Chiang, and M. Pawlak. Chinese character recognition via Gegenbauer moments. In *Proc. 16th International Conference on Pattern Recognition*, volume 3, pages 485–488, 2002.

[85] S. X. Liao and Q. Lu. A study of moment functions and its use in Chinese character recognition. In *Proc. 4th International Conference on Document Analysis and Recognition*, pages 572–575, August 1997.

[86] S. X. Liao and M. Pawlak. On image analysis by moments. *IEEE PAMI*, 18(3):254–266, 1996.

[87] L. Likforman-Sulem and M. Sigelle. Recognition of degraded characters using dynamic Bayesian networks. *Pattern Recognition*, 41(10):3092–3103, 2008.

[88] J. Mantas. An overview of character recognition methodologies. *Pattern Recognition*, 19(6):425–430, 1986.

[89] A. M Martinez and A. C. Kak. PCA versus LDA. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2):228–233, February 2001.

[90] J. Matas and K. Zimmermann. Unconstrained and text localization and recognition. In *Proc. 8th International IEEE Conference on Intelligent Transport Systems*, pages 572–577, September 2005.

[91] S. Mohanti. Pattern recognition in alphabets of Oriya language using Kohonen neural network. *International Journal of Pattern Recogn. Artif. Intell.*, 12:1007–1015, 1998.

[92] S. Mohanty, H. N. Dasbebartta, and T. K. Behera. An efficient bilingual optical character recognition (English-Oriya) system for printed documents. In *Proc. Seventh International Conference on Advances in Pattern Recognition*, pages 398–401, 2009.

[93] S. Mori, C. Suen, and K. Yamamoto. Historical review of OCR research and development. *Proceedings of the IEEE*, 80(7), July 1992.

[94] S. Mori and K. Yamamoto M. Yasuda. Research on machine recognition of hand-printed characters. *IEEE Transactions on Pattern Analysis and machine Intelligence*, 6(4), July 1984.

[95] R. Mukundan. Improving image reconstruction accuracy using discrete orthonormal moments. In *Proc. Intnl. Conf. on Imaging Systems, Science and Technology(CISST)*, pages 287–293, 2003.

[96] R. Mukundan. Some computational aspects of discrete orthonormal moments. *IEEE Image Processing*, 13(8):1055–1059, August 2004.

[97] R. Mukundan, S. H. Ong, and P. A. Lee. *Image and Vision Computing, NewZealand*, chapter Discrete Orthogonal Moment Features using Chebyshev Polynomials, pages 20–25. University of Canterbury, 2000.

[98] R. Mukundan, S. H. Ong, and P. A. Lee. Image analysis by Tchebichef moments. *IEEE Image Processing*, 10(9):1357–1364, September 2001.

[99] R. Mukundan and K. R. Ramakrishnan. Fast computation of Legendre and Zernike moments. *Pattern Recognition*, 28(9):1433–1442, 1995.

[100] R. Mukundan and K. R. Ramakrishnan. *Moment Functions in Image Analysis - Theory and Applications.* World Scientific, Singapore, 1998.

[101] G. Nagy. Chinese character recognition - a twenty five years retro-spective. In *Proc. Ninth International Conference on Pattern Recognition*, pages 109–114, 1988.

[102] G. Nagy. At the frontiers of OCR. *Proceedings of the IEEE*, 80(7), July 1992.

[103] G. Nagy and S. Seth. Modern optical character recognition. In F. E. Froehlich and A. Kent, editors, *The Froehlich/Kent Encyclopedia of Telecommunications*, pages 473–531. Marcel Dekker, New York, 1996.

[104] A. Negi, Bhagvati C, and B. Krishna. An OCR system for Telugu. In *Proc. Sixth International Conference on Document Processing*, pages 1110–1114, 2001.

[105] U. Pal and B. B. Chaudhuri. Printed Devnagari script OCR system. *Vivek*, 10:12–24, 1997.

[106] U. Pal and B. B. Chaudhuri. Automatic recognition of unconstrained on-line Bangla hand-written numerals. In T. Tan, Y. Shi, and W. Gao, editors, *Advances in Multimodal Interfaces; Lecture Notes on Computer Science (LNCS-1948)*, pages 371–378. Springer Verlag, 2000.

[107] U. Pal and B. B. Choudhuri. Indian script character recognition: a survey. *Pattern Recognition*, 37(9):1887–1899, September 2004.

[108] U. Pal and S. Datta. Segmentation of Bangla unconstrained handwritten text. In *Proc. Seventh International Conference on Document Analysis and Recognition*, pages 1128–1132, 2003.

[109] U. Pal, P. K. Kundu, and B. B. Chaudhuri. OCR error correction of an inflectional Indian language using morphological parsing. *Journal of Information Science Eng.*, 16:903–922, 2000.

[110] U. Pal, M. Mitra, and B. B. Chaudhuri. Multi-skew detection of Indian script documents. In *Proc. Sixth International Conference on Document Analysis and Recognition*, pages 292–296, 2001.

[111] R. Palaniappan, P. Raveendran, and S. Omatu. New invariant moments for non-uniformly scaled images. *Pattern Analysis & Applications*, 3(2):78–87, June 2000.

[112] S. Palit and B. B. Chaudhuri. A feature-based scheme for the machine recognition of printed Devanagari script. In P. P. Das and B. N. Chatterjee,

editors, *Pattern Recognition, Image Processing and Computer Vision*, pages 163–168. Narosa Publishing House, New Delhi, India, 1995.

[113] S. K. Parui, B. B. Chaudhuri, and D. Dutta Majumder. A procedure for recognition of connected hand written numerals. *International Journal of Systems Sci.*, 13:1019–1029, 1982.

[114] R. Plamondon and S. N. Srihari. On-line and off-line handwritten character recognition: A comprehensive survey. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(1):62–84, January 2000.

[115] W. K. Pratt. *Digital Image Processing.* John wiley & Sons Inc., New York, 1991.

[116] R. J. Prokop and A. P Reeves. A survey of moment-based techniques for unoccluded object representation and recognition. *Graphical Models and Image Processing*, 54(5):438–460, 1992.

[117] A. F. R. Rahman, R. Rahman, and M. C. Fairhurst. Recognition of handwritten Bengali characters: A novel multistage approach. *Pattern Recognition*, 35:997–1006, 2002.

[118] S. N. S. Rajasekaran and B. L. Deekshatulu. Recognition of printed Telugu characters. *Computer Graphics Image Processing*, 6:335–360, 1977.

[119] R. S. Rao and R. D. Sudhaker Samuel. On-line character recognition for handwritten Kannada characters using wavelet features and neural classifier. *IETE Journal of Research*, 46:387–392, 2000.

[120] T. M. Rath and R. Manmatha. Word spotting for historical documents. *International Journal on Document Analysis and Recognition (IJDAR)*, 9(2):139–152, 2007.

[121] P. Raveendran, S. Jegannathan, and S. Omatu. Classification of elongated and contracted images using new regular moments. In *Proc. IEEE International Conference on Neural Networks*, volume 6, pages 4154–4158, Orlando, FL, USA, July 1994.

[122] K. Ray and B. Chatterjee. Design of a nearest neighbor classifier system for Bengali character recognition. *Journal of Institute of Electronics and Telecom. Eng.*, 30:226–229, 1984.

[123] J. Rocha and T. Pavlidis. Character recognition without segmentation. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 17(9):903–909, 1995.

[124] S. Saharia, P. K. Bora, and D. K. Saikia. Representation of printed characters with Legendre moments. In *Proc. 5th International Conference on Advances in Pattern Recognition(ICAPR)*, pages 280–283, December 2003.

[125] S. Saharia, P. K. Bora, and D. K. Saikia. A comparative study on discrete orthonormal Chebyshev moments and Legendre moments for representation of printed characters. In *Proc. 4th Indian Conference on Computer Vision Graphics & Image Processing*, pages 491–496, 2004.

[126] S. Saharia, P. K. Bora, and D. K. Saikia. Fast computation of Legendre moments. In D. K. Bhattacharyya and S. M. Hazarika, editors, *Networks, Data Mining and Artificial Intelligence*, pages 170–177, 2006.

[127] S. Saharia, P. K. Bora, and D. K. Saikia. Improving character recognition accuracy of Tchebichef moments by splitting of images. In *Proc. Fifteenth National Conference on Communications(NCC 2009)*, pages 385–387, January 2009.

[128] S. Saharia, S.N.N. Pandit, and M. Borah. Orthogonal moments and principal component analysis in image reconstruction and pattern recognition. In *Proc. International Sympo. on Intelligent Robotic Systems*, pages 181–188, January 1998.

[129] T. Sano and T. Hananoi. Type H-8959 optical character reader. *Hitachi Review*, 54(12):1077–1082, 1972.

[130] K. K. Sarma. Modified hybrid feature set of Assamese character and Anglo-Assamese hand written numeral recognition. *International Journal of Imaging*, 1(A08), 2008.

[131] K. K. Sarma. Neural network based feature extraction for Assamese character and numeral recognition. *International Journal of Artificial Intelligence*, 2(S09), 2009.

[132] M. Schenkel and M. Jabri. Low resolution, degraded document recognition using neural networks and hidden Markov models. *Pattern Recognition Letters*, 19:365–371, March 1998.

[133] K. Sethi and B. Chatterjee. Machine recognition of constrained hand-printed Devnagari numerals. *Journal of Institute of Electronics and Telecom. Eng.*, 22:532–535, 1976.

[134] K. Sethi and B. Chatterjee. Machine recognition of constrained hand-printed Devnagari. *Pattern Recognition*, 9:69–76, 1977.

[135] I. Sheinberg. The INPUT2 document reader. *Journal of Pattern Recognition Society*, 2(3):167–173, September 1970.

[136] J. Shlens. *A Tutorial on Principal Component Analysis.* Institute for Nonlinear Science, University of California, San Diego, La Jolla, CA 92093-0402, 2 edition, December 2005.

[137] R. M. K. Sinha. *A Syntactic Pattern Analysis System and its Application to Devnagari Script Recognition.* PhD thesis, Electrical Engineering Department, Indian Institute of Technology, India, 1973.

[138] R. M. K. Sinha. Rule based contextual post-processing for Devnagari text recognition. *Pattern Recognition*, 20:475–485, 1987.

[139] R. M. K. Sinha and H. Mahabala. Machine recognition of Devnagari script. *IEEE Transaction on Systems Man Cybernatics*, 9:435–441, 1979.

[140] G. Siromony, R. Chandrasekaran, and M. Chandrasekaran. Computer recognition of printed Tamil characters. *Pattern Recognition*, 10:243–247, 1978.

[141] A. Som. *On Some Nonparametric Methods in Pattern Recognition.* PhD thesis, Jadavpur University, India, 1979.

[142] N. Srihari, H. Srinivasan, C. Huang, and S. Shetty. Spotting words in Latin, Devanagari and Arabic scripts. *Vivek: Indian Journal of Artificial Intelligence*, 16(3):2–9, 2006.

[143] S. N. Srihari and J. J. Hull. Character recognition. In S. C. Shariro, editor, *Encyclopedia of Artificial Intelligence*, pages 138–150. John Wiley, New York, 1992.

[144] W. Stallings. Approaches to Chinese character recognition. *Pattern Recognition*, 8(2):87–98, 1976.

[145] R. Sukhaswami, P. Seetharamulu, and A. K. Pujari. Recognition of Telugu characters using neural networks. *International Journal on Neural Systems*, 6:317–357, 1995.

[146] S. Sundaresan and S. S. Keerthi. A study of representations for pen-based hand writing recognition of Tamil characters. In *Proc. Fifth International Conference on Document Analysis and Recognition*, pages 422–423, 1999.

[147] G. Tausheck. Reading machine. U.S. patent 2026329, December 1935.

[148] M. R. Teague. Image analysis via general theory on moments. *J. of Opt. Soc. of America*, 70(8):920–930, 1980.

[149] C. H. Teh and R. T. Chin. On image analysis by method of moments. *IEEE Transation of Pattern Analysis and Machine Intelligence*, 10(4):496–513, July 1988.

[150] O. D. Trier, A. K. Jain, and T. Taxt. Feature extraction methods for character recognition-a survey. *Pattern Recognition*, 29(4):641–662, 1996.

[151] S. E. Umbaugh, Y. Wei, and M. Zuke. Feature extraction in image analysis. *IEEE Engineering in Medicine and Biology*, 16(4):62–73, 1997.

[152] G. Vamvakas, B. Gatos, N. Stamatopoulos, and S. J. Perantonis. A complete optical character recognition methodology for historical documents. In *The Eighth IAPR Workshop on Document Analysis Systems*, pages 525–532, September 2008.

[153] A. Venkataramana and P. Ananth Raj. Image watermarking using krawtchouk moments. In *Proc. International Conference on Computing: Theory and Applications(ICCTA '07)*, pages 676–680, March 2007.

[154] L. Wang and G. Healey. Using Zernike moments for the illumination and geometry invariant classification of multispectral texture. *IEEE Transsctions on Image Processing*, 7:196–203, 1998.

[155] R. Y. Wong and E. L. Hall. Scene matching with invariant moments. *Computer Graphics and Image Processing*, 8:16–24, 1978.

[156] W. H. Wong, W. C. Siu, and K. M. Lam. Generation of moment invariants and their use for character recognition. *Pattern Recognition Letters*, 16:115–123, 1995.

[157] P. T. Yap, R. Paramesran, and S. H. Ong. Image analysis by Krawtchouk moments. *IEEE Trans. Image Processing*, 12(11):1367–1377, November 2003.

[158] J. Ye. Least square linear discriminant analysis. In *Proc. 24th International Conference on Machine Learning*, pages 1087–1093, Corvalis, Oregon, 2007.

[159] H. Zhu, H. Shu, J. Liang, L. Luo, and J. Coatrieux. Image analysis by discrete orthogonal Racah moments. *Signal Processing*, 87:687–708, 2007.

[160] H. Zhu, H. Shu, T. Zia, L. Luo, and L. Coatrieux. Translation and scale invariants of Tchebichef moments. *Pattern Recognition*, 40:2530–2542, 2007.

[161] Indian languages. http://www.indianlanguages.com.

[162] Languages with official status in India. http://en.wikipedia.org/wiki/Official_languages_of_India.

Index