A Study of Gegenbauer and Tchebichef Moments for Telugu OCR

A Dissertation submitted to the University of Hyderabad
in partial fulfillment of the degree of

Master of Technology

in

Artificial Intelligence

By

S.RAJA LENIN BABU

09MCMI17

Department of Computer and Information Sciences

School of Mathematics & Computer Information Sciences
University of Hyderabad
Hyderabad-500046

Certificate

This is to certify that the dissertation entitled **A Study of Gegenbauer and Tchebichef Moments for Telugu OCR** submitted by **S.RAJA LENIN BABU** , bearing **Reg. No. 09MCMI17**, in partial fulfillment of the requirements for the award of Master of Technology in Artificial Intelligence, is a bonafide work carried out by him under my supervision and guidance.

The dissertation has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Dr. Atul Negi
Project supervisor,
Department of DCIS,
University of Hyderabad.

Head of Department,
Department of CIS,
University of Hyderabad.

Dean,
School of MCIS,
University of Hyderabad.

Declaration

I S.Raja Lenin Babu hereby declare that this Dissertation entitled "**A Study of Gegenbauer and Tchebichef Moments for Telugu OCR**" submitted by me under the guidance and supervision of **Dr. Atul Negi**, is a bonafide work. I also declare that it has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma.

Date:

Signature of the Student
Name : S.Raja Lenin Babu
Reg. No: 09MCMI17

Dedicated to,

My Family and Friends

Acknowledgements

I would like to express my sincere gratitude to **Dr. Atul Negi**, my project supervisor, for valuable suggestions and keen interest through out the progress of my course of research.

I would like to thank **Prof. B L Deekshatulu**, **Prof. Arun Agarwal**, **Prof. Chakravarthy Bhagvati** for their valuable suggestions through out the project.

I am grateful to **Prof. C R Rao**, Head, DCIS for providing excellent computing facilities and a congenial atmosphere for progressing with my project.

I would like to take this opportunity to thank **P. Pavan Kumar**, a Ph.d student in OCR lab for giving immense support and helping me through out the completion of project.

At the outset, i would like to thank **The University of Hyderabad** for providing all the necessary resources for the successful completion of my course work.

At last, but not the least i thank my classmates and other students of DCIS for their physical and moral support.

With Sincere Regards,

S.RAJA LENIN BABU .

Abstract

Telugu OCR system has various modules like pre-processing (binarization, skew detection and correction etc.), Line and word segmentation, classification and finally post-processing. Each module has its impact on the performance of the overall system and hence an improvement in each module immediately reflects in the overall system performance. Despite success of systems in several input patterns, problems that involve recognising closely related patterns remains difficult. Most of the Classifiers in the OCR system fails to correctly classify an input pattern that belongs to the Confusion pair (eg: *na* and *va*).

In our work, we focus on character recognition module using moment features. Moment features have been applied in pattern recognition systems since the moment method was developed.We are using Gegenbauer and Tchebichef moments in our work for character recognition. Previously other moments like Zernike moments are used in Telugu Character recognition, but found that procedure for Fringe distance is successful when compared with the moments.

In this work we are experimenting with Gegenbauer and Tchebichef moments, which are useful in other Optical Character Recognition works.

Contents

List of Figures

Chapter 1

Introduction

Optical Character Recognition is presently one of the important enabling technologies for the progress of language oriented work. The transformation of paper media text into the searchable and computer revisable format gives research in the field of language technologies a great boost. OCR is being used in the context of language technology research for creation of text corpora. Office automation and content creation activities are the important areas of OCR applications.

Telugu is the language spoken by more than 100 million people of South India. Telugu has a complex orthography with a large number of distinct character shapes, estimated to be around 10,000, composed of simple and compound characters formed from 16 vowels, called *acchulu*, and 36 consonants, called *hallulu*. In addition several semi-vowel symbols, called *maatra*, are used in conjuction with *hallus* and half consonants, called *voththulu* are used in consonant clusters.

OCR system for printed Telugu script has been built previously[9]. However, the off-line handwritten Telugu script segmentation is a challenging problem due to the text of two consecutive lines may touch or overlap. The aim of this project is to investigate line segmentation approaches which handles the complex nature of Telugu script.

1.1 Problem Statement

To study the character recognition approaches which handles the complex nature of Telugu script.

1.2 Motivation

Following drawbacks in the existing Telugu OCR system motivated me to study character recognition approaches for Telugu OCR system (DRISHTI).

- Problems that involve recognizing closely related patterns remains difficult.Most of the Classifiers in the OCR system fails to correctly classify an input pattern that belongs to the Confusion pair (eg: na and va).
- Performance and the accuracy level in the Character recognition module is less, which when increased will immediately reflects in the overall system performance.

1.3 Overview of Project Report

In this report the following chapters explain about the proposed work, mathematical formulae and the results obtained. Chapter 2 explains the background of the Telugu OCR, all the steps that are in the OCR.Then in chapter 3,it gives the information regarding the proposed work and the mathematical formulae.And in the chapter 4, shows the results that are obtained. Conclusion and Future work is in Chapter 5.

Chapter 2

Background

This chapter gives a brief introduction to Document Analysis and Recognition, OCR, history, system design and applications of OCR are discussed briefly. Telugu script is described briefly and an approach for the OCR algorithmic is given.

2.1 Document Analysis and Recognition

Document analysis and recognition is concerned with the automatic interpretation of images of printed and handwritten documents, including text, pictures, graphics, engineering drawings, maps, music scores, etc. It aims at the transformation of data presented on paper into a computer-revisable form, where where the pixel representation of a scanned document is converted into symbolic entities that are appropriate for the intended kind of computerized information processing. Research in this field descends in an unbroken tradition from the earliest experiments in computer vision, and remains distinguished by close and productive ties between the academic and industrial communities [11]. While the difficulty of its characteristics continues to stimulate basic research, advanced techniques in this area support a rapidly growing industry.

Document analysis and recognition is a growing field with a wide spectrum of applications. For example, it can be used to convert paper-based engineering drawings into CAD-compatible form, archive and retrieve documents such as journals and magazines, reduce the storage of mixed text and graphics documents, and in general, do intelligent interpretation of document image information.

2.2 Optical Character Recognition (OCR)

OCR has been a popular focus of patters recognition research since at least the 1960's. The ready availability of image samples and the continuing challenge of commercially viable recognition has kept OCR research ongoing [12].The samples are the printed documents, such as newspaper, books, and magazines. The importance of text in human transactions made automatic recognition a practical significance. There are various types of paper documents printed from printer, type writer or photocopied documents or the combination of one or more types of the above. Documents from printers or typewriters usually will have a limited variation of size and fonts.

The process of document recognition involves the acquisition of document images, segmentation of the document images into text and rows. Followed by further segmentation of each row into individual words. Now the recognition problem can be solved in two ways either at the word level or at the character level. If the document is type-written or printed then the best method is to do recognition at character level. After recognition of individual letters, the letters are integrated to generate words and words into sentences and finally the whole document information is retrieved. The system that performs the above task is called Optical Character Recognizer and the process is called Optical Character Recognition.

Optical Character Recognition is the process of converting the scanned images into a computer processable format. Depending upon the applications, OCR systems are built. For printed documents recognition, a simple OCR that can recognize isolated characters is sufficient. But for applications like cheques reading, visiting card verification, OCR system should be able to recognize cursive script or printed text as well as graphics blocks in the document.

2.2.1 Brief History of OCR

OCR was the first AI system for character recognition [10]. In the early 1960's OCR digital pattern recognition system were conceived to recognize printed char-

acters, and they limited the way we thought human beings perceived to recognize printed characters. In the 1970's, it became possible to electronically store an image of a document and to character-recognize from the stored image. That allowed a great increase in the intelligent of those OCR systems. OCR software has become increasingly sophisticated in its ability to recognize text, thus ensuring a greater accuracy. What were stumbling blocks in the past, such as typographical and formatting complexities (e.g. bold, italics, font size, tables) are being overcome with powerful recognition features that most software now includes. However, accuracy rates are only in the high 90 percentiles, still requiring clean-up in the post-OCR phase.

2.2.2 OCR System Design

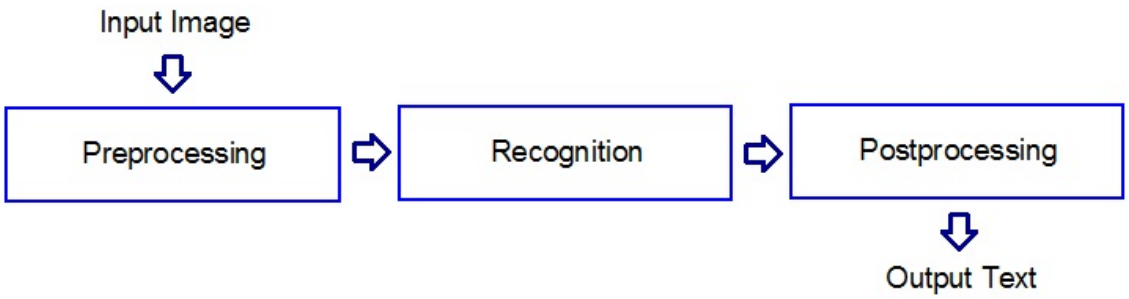


Figure 2.1: Block diagram of OCR system

A typical OCR, as shown in Fig. `fig:ocr-block?`, contains three phases- preprocessing, recognition and post processing. The pre-processing phase includes binarization of the input document image to separate the print and background objects, noise removal, skew correction, extraction of layout information etc. In the recognition phase each character in the document image is recognized. Two alternative techniques that can be used in this phase are template matching and feature extraction. In template-matching, each character in the input image as seen by OCR is compared against a set of templates and the code of the template that best matches it is output. In feature extraction method general features like open areas, closed shapes, lines, intersections etc. are examined in the output character based on which the character is recognized. The post-processing phase includes conversion of the output into any standard text-encoding scheme, restoring the layout, detection and

correction of errors made in the recognition phase.

A research prototype system of recognition phase to recognize Telugu character was designed and implemented in [10]. It recognizes each word in the input image and output ISCII (Indian standard code for information interchange, a common standard text encoding scheme for all Indian languages) code for that word. Preprocessing and post-processing phases are not implemented. Little importance was given to speed and efficiency. Concepts like binarization, matching criteria etc, which affect the recognition rate, were not investigated in depth. The algorithm did not consider some peculiar properties of Telugu script, which resulted in ambiguity while converting OCR glyph codes to ISCII. It also put limits on the size of the image that can be processed.

2.2.3 Applications of OCR

The applications of OCR spread in various kinds of fields. OCRs can be divided broadly into two types depending on their applications. They are specific reader and general purpose page reader. Task specific readers are designed to handle specific types of documents like bank cheques, bills, applications forms etc. In this case, the format of the input documents is fixed and the OCR has to recognize a particular areas in the document . Such systems emphasize on high throughput rates and low error rate. Many of them recognize both handwritten and machine printed text.

General purpose page readers are designed to handle a range of documents such as business letters, technical writings and newspapers. These systems capture an image of a document page and separate the page into text region and non text regions. Non text regions such as graphics and line drawings are often saved separately from the text and associated recognition results. Text regions are segmented into lines, words and characters and characters are passed to the recognizer. Recognition results are output in a format that can be post processed by application software. Most of these page readers can read machine written text but only a few can read hand printed alphanumeric.

Some of the general techniques followed in OCR are described in [10].

2.3 Telugu Script

Telugu is a phonetic language spoken by more than 100 million people in south India. Telugu script is written from left to right, with each character representing a syllable. Telugu script has 16 vowels, which are called *Achchulu* and 36 consonants, which are called *Hallulu*. These are shown in Fig. 2.2 and Fig. 2.3 respectively. From the figure, it can be seen that all of them are composed of circular segments with different radii.

Figure 2.2: Vowels of Telugu script
[fig:achchulu]

Figure 2.3: Consonants of Telugu script
[fig:hallulu]

In English vowels are used in two ways.

1. They are used to produce their own sounds. For example, in the word 'ink' the vowel 'I' is used to produce its own sound.

2. They are used to modify the sound of a consonant. For example, in the word 'kill' the same vowel 'I' is used to modify the sound of consonant 'k'.

But in Telugu vowels that are used for the first purpose are not used for the second purpose. Instead, Telugu consist of a special symbol called *Maatra*, corresponding to each vowel, which are attached to *hallus* to modify their sound. *Maatras* corresponding to each vowel in Fig. 2.2 are shown in Fig. 2.4. In Fig. 2.5 each maatra has been attached to the first *hallu*. These sequence of characters obtained by adding *maatra* to *hallus*, are called *Guninthalu* in Telugu. From the figure it can be seen that some *maatras* are placed at the top, some at bottom right and some at bottom part of the consonant. Same *maatra* can be attached at different positions for different *hallus* and same *maatras* can occur in different shapes depending on the *hallu* to which it is attached.

Figure 2.4: *Maatras* corresponding to each vowel
[fig:gunintham]

Figure 2.5: *Maatras* corresponding to each vowel attached to first *hallu*
[fig:ka-gunintham]

Besides all of these, there are half consonants called *Vottus*, corresponding to each consonant. These *vottus* are placed at a distance at the bottom or right corner of *Hallulu*. The vottus for the respective Hallulu in Fig. 2.2 are shown in Fig. 2.6

A character is said to be simple if it is an *achchu* or a *hallu* alone with a *maatra*. A character is said to be compound if it is a *hallu* with a *maatra* and with any number of *vottus* . In practice, characters with more than two *vottus* are very rare.

Figure 2.6: The *vottus* for the respective *Hallulu*
[fig:vottu]

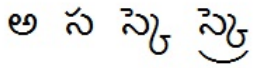


Figure 2.7: Example of simple and complex characters in Telugu Script
[fig:simple-complex]

All the *achchus*, *hallus*, *maatras*, *vottus* and *hraswaksharas* together roughly provide 130 basic orthographic units, which are referred as glyphs that are combined together in different ways to represent all the frequently used syllables.

2.4 DRISHTI OCR

An OCR system for Telugu [9], DRISHTI, was developed in University of Hyderabad. It's a complete Optical Character Recognition system for Telugu language. The system is tested with a number of different fonts provided by C-DAC[1] and Modular Infotech, and on several popular novels, laser and desktop printer generated pages and books. Drishti is the first comprehensive OCR system for Telugu. A truthing

[1]Center for Development and Advanced Computing

tool with facilities for creating ground truth information, and to review the ground truth against image data is also implemented. Such truthing tools are extremely important in objective and quick evaluation of OCR system performance.

2.4.1 The Complete Algorithm

The complete Telugu OCR algorithm is given below.

Algorithm 1: The Complete Algorithm

1. Read an input binary image.
2. Segment the image into words.
3. Extract the connected components from each word.
4. For each component
   (a) Normalize size to match stored templates.
   (b) Compute fringe distance map.
   (c) Compute fringe distance of the fringe map from all templates.
   (d) Output template with smallest fringe distance.
   (e) Convert template code to ISCII/Unicode.
5. Store ISCII/Unicode output in a file.

The algorithm is shown as a schematic block diagram in Fig. 2.8

1) Binarization

The input grayscale document image is binarized so that all black pixels belong to characters and white pixels belong to background. A global threshold of 150 is used to binarize the image.

2) Word segmentation

The binarized image is segmented into words using RLSA (Run length smoothing algorithm). It connects adjacent black components that are with in a distance, t. The algorithm is applied to a binary image, first along rows and then along columns. The threshold, t is taken as half of font size, so that characters within a word are

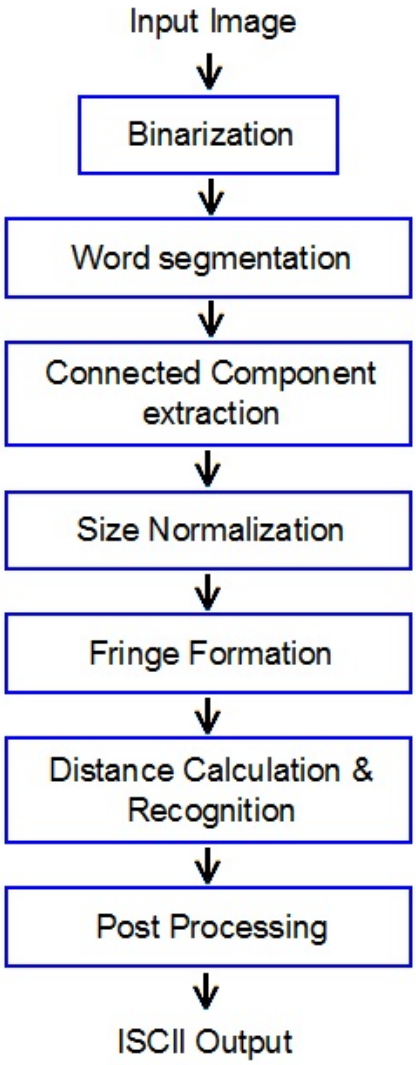


Figure 2.8: Schematic block diagram of Telugu OCR
[fig:drishti-block]

connected but adjacent words are not connected. On the result, the algorithm is applied along columns so that the *maatras* and *vottus*, which lie above and below the base character, are connected to the character but two words in adjacent rows are not connected. Threshold equal to $\frac{1}{3}$ of interline space is used. Font size and the interline space are found using projection profiles and crossing counts.

Connected components algorithm is applied on the result of RLSA so that each word is distinguished with a label. From the output of the connected components algorithm, the coordinate of the minimum bounded rectangle for each word are

found. Sub images of the binaries images are created using the word MBRs. These sub images contain one word per image. Each of these word images is sent to next step.

3) Connected Component Extraction

From each word all the connected components are extracted as follows. The input word image is processed along a vertical scan line, the scan line moving from left to right. When a black pixel comes in the scan line, its position is stored and all its 8 connected pixels are checked to see if they are black. For each of these black pixel is again their positions are stored and all 8 connected pixels are checked and this process is repeated till there are no black pixels in the 8 connected region. Each time when the position of black pixel is stored, that pixel is made white so that it is not processed again. Now from the stored pixel positions a new image is created which gives the connected component. This way the entire connected component in the word is obtained as separate images. Each of these images is to size normalization.

4) Size normalization

In this step the size of the connected components is normalized to template size. A simple linear scaling method is used. Let the size of the input component be IxJ and the size of the template is LxM . (This is fixed as 32*32). Then for each black pixel $p(i, j)$ in the input component the position in the transformed image $q(l, m)$ is computed as follows.

$$l = round(\frac{i*L}{I})$$
$$m = round(\frac{j*M}{J})$$

Where $round(x)$ is the nearest integer to x. All the remaining positions of the transformed image are background pixels.

Another approach called nonlinear normalization [3] was also used to normalize the input component to template size. This is based on projection profiles.

5) Fringe formation

In this step the fringe distance map of the scaled image is computed. Fringes and fringe distance maps are described in previous section. The algorithm for fringe formation is given below.

Algorithm 2: Algorithm for fringe formation

```
foreach black pixel (i, j) do
    FRING (i, j) ← 0;
    Add (i, j) to list;
    while while there are unprocessed pixels do
        fringeno++;
        foreach pixel P in list do
            foreach processed 8 connected neighbor (i, j) of P do
                FRING (i, j) ← fringeno;
                Add (i, j) to newlist;
        list ← newlist;
```

The fringe array is written to a file and the file is sent for comparison against all template fringes.

6) Distance calculation & Recognition

The distance between input fringe map and all the template fringe maps is calculated as described in previous section and the glyph code of the nearest template is written to output file. The output file contains a sequence of glyph codes for all the words in the input image, one word per line.

7) Post processing

In this step for each line in the output file, the subsequences of glyph codes that make a character are identified and the ISCII code of corresponding character is written in the new output file. ISCII code for 'space' character is inverted after each word.

Chapter 3

Character Recognition using Moment based methods

3.1 Introduction

3.1.1 Why moment based methods used??

Moment features have been applied in pattern recognition systems since the moment method was developed. In our work, a set of moment features extracted from the Gegenbauer moment method for Chinese character recognition [4] is followed. Compared with the results based on other moment methods, the Gegenbauer moments can provide a modest improvement in terms of recognition for those Chinese characters that are very similar in shapes.Gegenbauer moment method can supplement the existing Chinese character recognition techniques based on local structure features. Discrete orthogonal moments such as Tchebichef moments [5] have been successfully used in the field of image analysis. However, the invariance property of these moments has not been studied mainly due to the complexity of the problem. Conventionally, the translation and scale invariant functions of Tchebichef moments can be obtained either by normalizing the image or by expressing them as a linear combination of the corresponding invariants of geometric moments. In our work, we followed the approach [5] that is directly based on Tchebichef polynomials to derive the translation and scale invariants of Tchebichef moments. Both derived invariants are unchanged under image translation and scale transformation. The performance of the descriptors is evaluated using a set of binary characters.

3.2 Background

Previously Hu's moments [8] and Zernike moments [7] are used in Telugu character recognition. Hu first demonstrated the utility of moment features in a recognition experiment by using a set of 26 capital English letters as input patterns. In the two-dimensional moment feature space, all points representing each of the characters were fairly distinct except those of M and W. To describe an image with moment features means that the global properties of the image are used rather than the local ones. Compared with the set of English letters, that of Telugu characters is much larger and more difficult to classify. In addition, many of Telugu characters are similar in shapes but very different in meanings. However, two Telugu characters that have very similar local structures are not necessarily close in terms of the global structures, which can be represented effectively by the moment features. Zernike moments were previously used in Telugu Character recognition, but were not effective.The fringe distance methods are very effective when compared to the Zernike moments.
Advantages of using Zernike moments are :

- Easy image reconstruction ability,
- Selection of the required maximum order can be decided by image reconstruction results.

Disadvantages of using Zernike moments are:

- Zernike moments themselves are only invariant to rotation;
- the image need to be normalized first to obtain scale and translation invariance.

Its has been implemented and have seen that Zernike moments are not effective for TELUGU Character recogntion.

3.3 Chinese Character Recognition via Gegenbauer Moments

3.3.1 Introduction

Optical Character Recognition (OCR) systems for printed Chinese characters have been designed to automate the process of inputting Chinese documents with decent performances. However, one of the main challenges for the existing systems is to recognize the Chinese characters that are very close in shapes. Unlike many of other languages, the structures and components of Chinese characters often convey information on the meanings and pronunciations of those characters. These specific information is used by many Chinese character feature extraction methods. Quite often, however, Chinese characters that are different in meanings are very similar in shapes thus making classification very difficult.

The recognition rate of an OCR system largely depends on the features employed to differentiate one character from another. Although more extracted features will provide the higher correction rate for a Chinese character recognition system, considering the large set of Chinese characters, only a very small number of features can be used in order to obtain results in a reasonably short time. Consequently, the selection of feature extraction methods becomes the single most important factor in achieving high recognition rate. Many feature extraction methods used in the existing Chinese character recognition systems, such as the stroke feature method and the feature point method, etc.use the local features of the characters. However, these methods are less effective to recognize Chinese characters that are very close in shapes. The moments of an object have been extracted and used as the features in pattern recognition. The moment method is different from other feature extraction methods used in Chinese character recognition [4] because the moment method captures global properties of characters rather than the local ones.

Moment features can represent two Chinese characters uniquely no matter how close the two characters are in terms of their local features. This uniqueness nature of moments makes the method of moments an ideal candidate in Chinese character recognition. In this paper, a Chinese character recognition system based on a newly developed moment function, Gegenbauer moment [4], is followed. We apply the feature extraction functions, with Gegenbauer moments to the 6,763 Chinese characters

defined in the Chinese standard GB2312. Results show that the new system can provide the improved performances and perform particularly well in recognizing the characters of close shapes, which makes the method a good complement for other Chinese character recognition techniques based on the local features.

3.4 Mathematical Formulation

3.4.1 Gegenbauer Moments

The Gegenbauer polynomials [4] is a class of orthogonal polynomials on the interval [-1,1] characterized by a single parameter λ that allows to change the form of the polynomial. Hence let Gn (x; λ) denote the Gegenbauer polynomial of order n with the parameter λ, which can be any real number satisfying the restriction of λ >-0.5. The Gegenbauer polynomial system is related to the Jacobi polynomials

$$G_n(x;\lambda) = \frac{\Gamma(\lambda+1/2)}{\Gamma(2\lambda)}\frac{\Gamma(n+2\lambda)}{\Gamma(n+\lambda+1/2)}P_n^{(\lambda-1/2,\lambda-1/2)}(x) \tag{3.1}$$

$\Gamma(x)$ is the Gamma function. Note that when λ =0.5, Gn(x; 0.5) becomes the Legendre polynomials $Pn(x)$.
with the normalizing constant,

$$C_n(\lambda) = \frac{2^{2\lambda}\Gamma^2(\lambda)}{2\pi}\frac{n!}{\Gamma(n+2\lambda)}(n+\lambda), \lambda \neq 0 \tag{3.2}$$

Let $f(x, y)$ be an image function, we can define the (n, m) Gegenbauer moment in a two-dimensional space as follows,

$$A_{p,q}(\lambda) = C_p(\lambda)\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} f(u,v)G_p(u;\lambda)G_q(v;\lambda)(1-u^2)^{(\lambda-1/2)}(1-v^2)^{(\lambda-1/2)}dudv \tag{3.3}$$

where λ>-0.5.

3.4.2 Moment Features

Hu first demonstrated the utility of moment features in a recognition experiment by using a set of 26 capital English letters as input patterns. In the two-dimensional moment feature space, all points representing each of the characters were fairly distinct except those of M and W. To describe an image with moment features means that the global properties of the image are used rather than the local ones. Compared with the set of English letters, that of Telugu characters is much larger and more difficult to classify. In addition, many of Telugu characters are similar in shapes but very different in meanings. However, two Telugu characters that have very similar local structures are not necessarily close in terms of the global structures, which can be represented effectively by the moment features. To compare the performances of the Gegenbauer moment [4] based new system, the same set of four feature functions of lower order moments

$$f_1 = A_{2,0}(\lambda) + A_{0,2}(\lambda) \tag{3.4}$$

$$f_2 = \sqrt{(A_{2,0}(\lambda) - A_{0,2}(\lambda))^2 + 4A_{1,1}(\lambda)} \tag{3.5}$$

$$f_3 = \sqrt{(A_{3,0}(\lambda) - 3A_{1,2}(\lambda))^2 + (3A_{2,1}(\lambda) - A_{0,3}(\lambda))^2} \tag{3.6}$$

$$f_4 = A_{3,0}(\lambda) + A_{0,3}(\lambda) \tag{3.7}$$

is adopted in this research. The Gegenbauer moment features would represent all Telugu characters in the image plane (x,y). A point $(f1, f2, f3, f4,)$ in this four-dimensional moment feature space represents one Chinese character. In our experiments, we change the values of parameter $\lambda$to study how the recognition results will be affected by different λ s.

Example showing the calculated feature values using Gegenbauer Moments

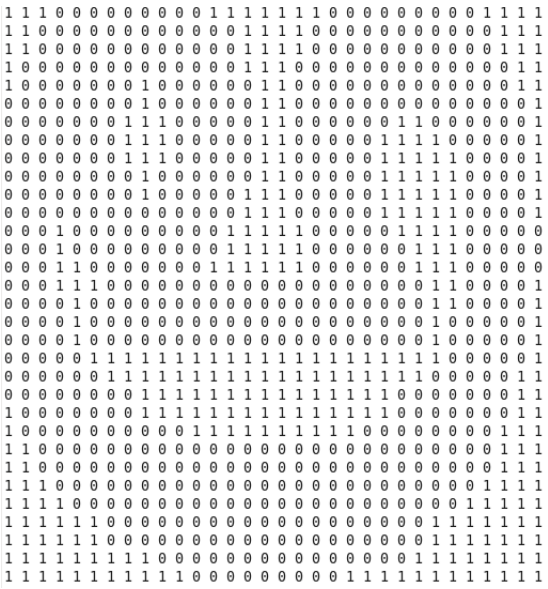


Figure 3.1: Sample image of AA

Table 3.1: Values

A(2,0)	485986.25000
A(0,2)	96919.00000
A(3,0)	28048012.00000
A(0,3)	4040768.5000
A(1,2)	4989845.00000
A(2,1)	7997082.00000
A(1,1)	144748.500000

The four feature values for the above character are:

Table 3.2: Feature Values

f1	582905.250000
f2	389068.000000
f3	23855150.000000
f4	32088780.000000

3.5 Tchebichef Moments

3.5.1 Introduction

Since Hu [8] first introduced the moment invariant, moments and moment functions have been widely used in the fields of image analysis and pattern recognition. Hus moment descriptors are invariant with respect to translation, scale and rotation of the image. However, the kernel function of geometric moments of order $(p + q)$, is not orthogonal, thus the geometric moments suffer from the high degree of information redundancy, and they are sensitive to noise for higher-order moments. Zernike and Legendre moments were later introduced by Teague who used the corresponding orthogonal polynomials as kernel functions. Another related orthogonal moments, denoted as pseudo-Zernike moments , was derived based on the basis set of pseudo-Zernike polynomials. These orthogonal moments have been proved to be less sensitive to image noise as compared to geometric moments, and possess better feature representation ability . The use of discrete orthogonal polynomials as basis functions for image moments eliminates the need for numerical approximations, and satisfies perfectly the orthogonality property in the discrete domain of image coordinate space. This property makes the discrete orthogonal moments superior to the conventional continuous orthogonal moments in terms of image representation capability. Recently, the rotational invariants of Tchebichef moments were proposed by Mukundan [5] . He constructed the rotational invariants using the one-dimensional Tchebichef polynomial along radial direction and a circular-harmonic function along the angular direction. To the best of our knowledge, until now, no report has been published on how to derive the translation and scale invariants of discrete orthogonal moments. We followed a approach [5] to derive the translation and scale invariants of Tchebichef moments based on the corresponding polynomials. This approach eliminates the requirement of calculating the normalization parameters of the shifted and scaled image, or utilizing other indirect methods to achieve the translation and scale invariance.

3.6 Mathematical Formulation

3.6.1 Tchebichef moments

In this section, we first review the theory of Tchebichef moments , and then give a brief description on how to derive the translation and scale invariants of Tchebichef moments from the geometric moments [5]

3.6.2 Tchebichef polynomials

The discrete Tchebichef polynomial of order n is defined as,

$$t_n(x) = (1-N)_n {}_3F_2(-n, -x, 1+n; 1-N; 1) \tag{3.8}$$

$n, x = 0, 1.....N-1$
where ${}_3F_2(.)$ is the generalized hypergeometric function given by

$${}_3F_2(a1, a2, a3; b1, b2; z) = \sum_{k=0}^{\infty} \frac{(a_1)_k (a_2)_k (a_3)_k}{(b_1)_k (b_2)_k} \frac{z^k}{k!} \tag{3.9}$$

and NXN is the image size, $(a)_k$ is the Pochhammer symbol given by

$(a)_k = a(a+1)(a+2).....(a+k-1), k \geq 1$ and $(a)_0 = 1$

We use the following scaled Tchebichef polynomials,

$$\tilde{t}_n(x) = \frac{t_n(x)}{\beta(n, N)} \tag{3.10}$$

where $\beta(n, N)$ is a suitable constant which is independent of x.

$$\beta_n(x) = \sqrt{\rho(n, N)} \tag{3.11}$$

the squared-norm of the scaled polynomials is,

$$\tilde{t}(n,N) = \frac{\rho(n,N)}{\beta(n,N)^2} \tag{3.12}$$

The two-dimensional (2D) Tchebichef moment of order n+m of an image intensity function, f(x, y), is defined as,

$$T_{nm} = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} \tilde{t}_n(x)\tilde{t}_m(x)f(x,y) \tag{3.13}$$

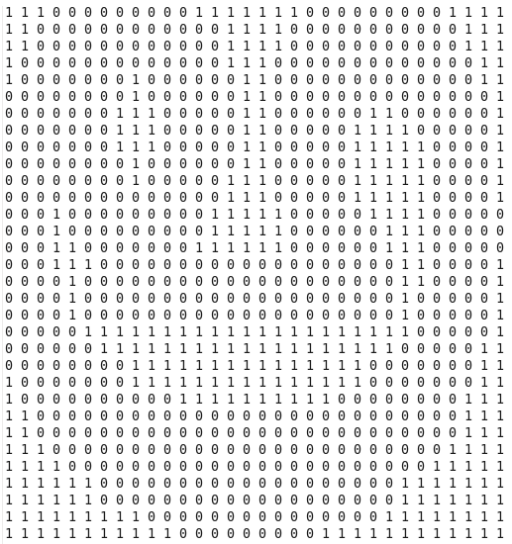


Figure 3.2: Sample image of a letter

The feature value computed for the above character is:

Table 3.3: Values

$t_n(x)$	76.679847
$t_m(x)$	91.515861

Tnm = 226764.646553

Chapter 4

Experimentation and Results

In this section,we discuss about calculating the feature values using Gegenbauer polynomials and Tchebichef polynomials. In chapter 3 already we have discussed about the mathematical formulas of Gegenbauer polynomials and Tchebichef polynomials. We maintain the feature values of 772 samples in the database. We calculated the feature values of 772 values using both the Gegenbauer polynomials and Tchebichef polynomials. All the 772 samples are binary inputs,where object pixels are 0 and background pixels are 1. Giving the 772 binary samples as inputs, and storing them in the database.
Examples of the binary samples.....

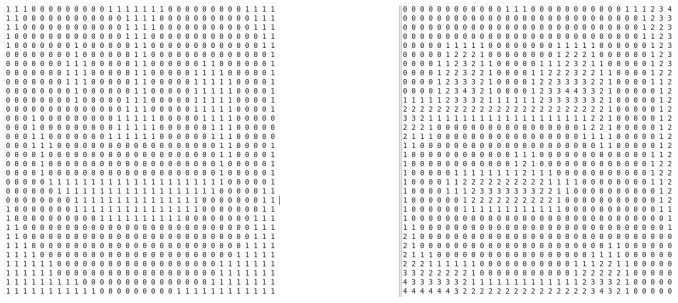
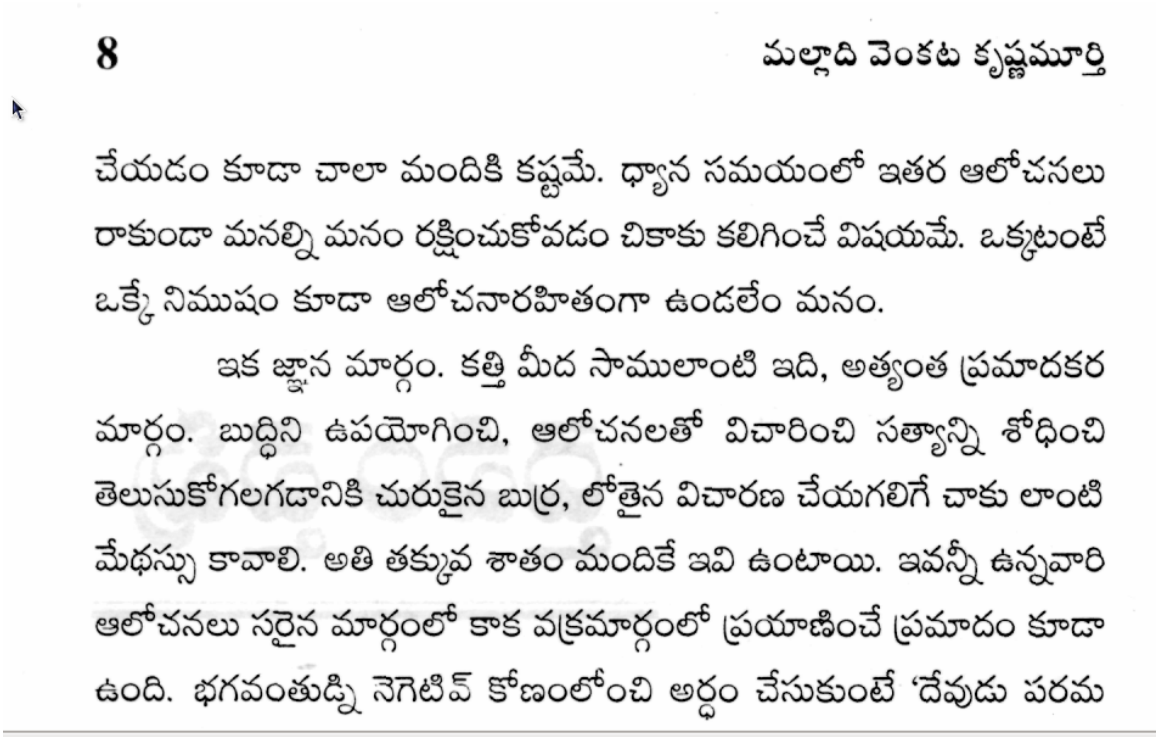


Figure 4.1: Binary Samples

4.1 Results

There are several steps included in calculating the feature values.
1.Giving the images as input to the DRISHTI OCR from the book "Daivam vaipu".
2.Calculating the feature values for all the connected components.
3.Finding the Euclidean distance between the connected component feature values and the databse feature values.
4.Returning the connected component which has minimum distance as output.
Example of input images from "DAIVAM VAIPU" book.



8 మల్లాది వెంకట కృష్ణమూర్తి

చేయడం కూడా చాలా మందికి కష్టమే. ధ్యాన సమయంలో ఇతర ఆలోచనలు రాకుండా మనల్ని మనం రక్షించుకోవడం చికాకు కలిగించే విషయమే. ఒక్కటంటే ఒక్కే నిముషం కూడా ఆలోచనారహితంగా ఉండలేం మనం.

ఇక జ్ఞాన మార్గం. కత్తి మీద సాములాంటి ఇది, అత్యంత ప్రమాదకర మార్గం. బుద్ధిని ఉపయోగించి, ఆలోచనలతో విచారించి సత్యాన్ని శోధించి తెలుసుకోగలగడానికి చురుకైన బుర్ర, లోతైన విచారణ చేయగలిగే చాకు లాంటి మేధస్సు కావాలి. అతి తక్కువ శాతం మందికే ఇవి ఉంటాయి. ఇవన్నీ ఉన్నవారి ఆలోచనలు సరైన మార్గంలో కాక వక్రమార్గంలో ప్రయాణించే ప్రమాదం కూడా ఉంది. భగవంతుడ్ని నెగెటివ్ కోణంలోంచి అర్థం చేసుకుంటే 'దేవుడు పరమ

Figure 4.2: Image from Book *DAIVAM VAIPU*

Connected components generated from the above page.....

Figure 4.3: Connected Components generated from the above page.

List of characters that are classified correctly using Gegenbauer moments are in the below table:

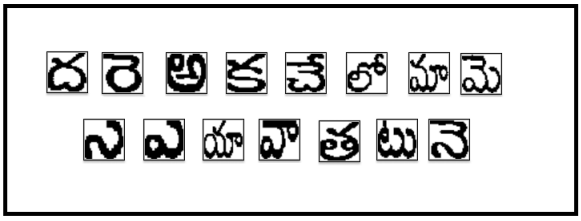


Figure 4.4: Gegenbauer correctly classified characters

List of characters that are misclassified using Gegenbauer moments are in the below table:

CHARACTERS	DISTANCE	CLASSIFIED CHARACTER
	36745.405573	
	434.927948	
	1110.885603	
	2258.903575	
	25721.856644	
	14871.572130	
	159418.984346	
	52854.04465	
	34353.773078	
	904.101066	
	19489.882560	

Figure 4.5: Gegenbauer misclassified characters

List of characters that are classified correctly using Tchebichef moments are in the below table:

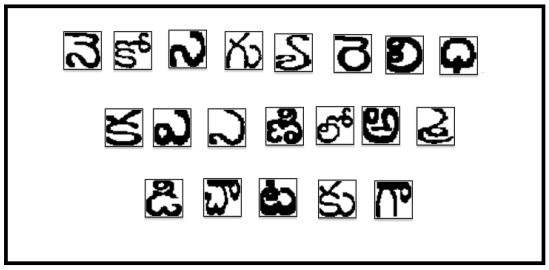


Figure 4.6: Tchebichef correctly classified characters

List of characters that are misclassified using Tchebichef moments are in the below table:

CHARACTERS	DISTANCE	CLASSIFIED CHARACTER
	2736.277365	
	1185.373874	
	761.312620	
	3751.380800	
	149529.415250	
	541.938048	
	6800.104418	
	144.996319	
	1522.768388	
	33761.170766	
	1221.213660	

Figure 4.7: Tchebichef misclassified characters

4.2 Scaling the image size

We have also did the experimentation by scaling the image size to 64X64. Initially the size of the image is 32x32. But by applying the scaling on the images the classification is not so effective.

Some of the characters that are misclassified are shown in the below table.

CHARACTERS	DISTANCE	CLASSIFIED CHARACTER
	2298.253160	
	488.851315	
	747.653606	
	7018.5000972	
	583.674014	
	655.145516	
	2934.463924	
	144.996319	
	1413.800590	
	1119297.636695	
	12426.782440	

Figure 4.8: Characters classified after Scaling the image

4.3 Zoning Approach

4.3.1 Experimentation 1:

Previously we had applied our moment features on the entire image. But in this Zoning approach, we had divided the image into four equal zones as shown in the figure 4.9

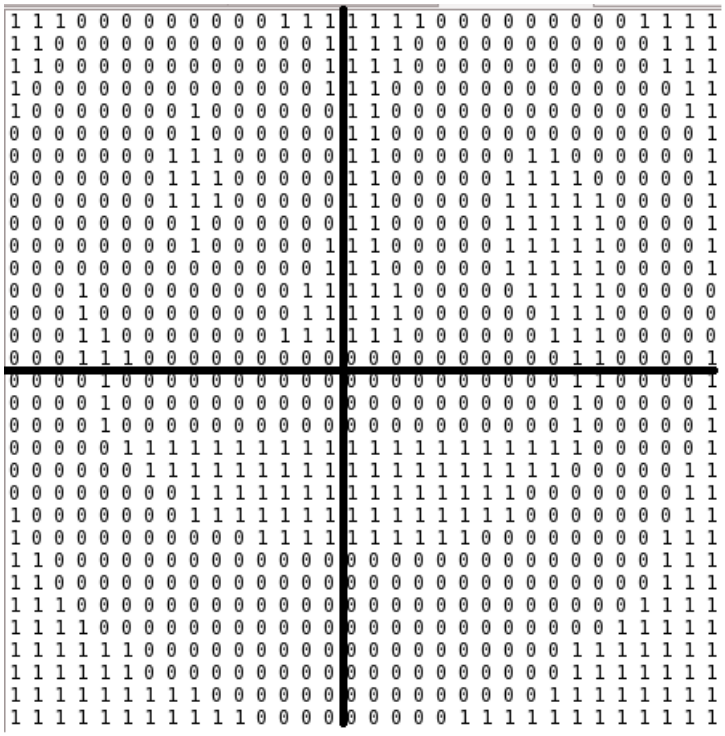


Figure 4.9: zoning

Then applying our moments on each zone rather than applying on the entire image, as shown in the figure 4.10

But by applying method, it is not effective in character recognition. It was experimented and found not successful.

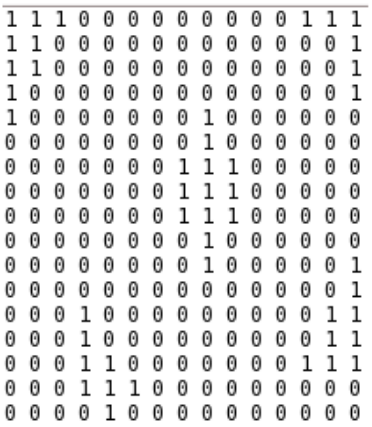


Figure 4.10: One of the *zone* from the above image

CHARACTERS	DISTANCE	CLASSIFIED CHARACTER
	411.809488	
	1518.747384	
	167.213620	
	3552.803330	
	122995.514220	
	111.998025	
	8812.441018	
	9663.761885	
	1725.863788	
	173.661706	
	2219.569780	

Figure 4.11: Characters misclassified after applying Zoning

4.3.2 Experimentation 2:

We also experimented by dividing the image into 16 zones, as shown in the below figure

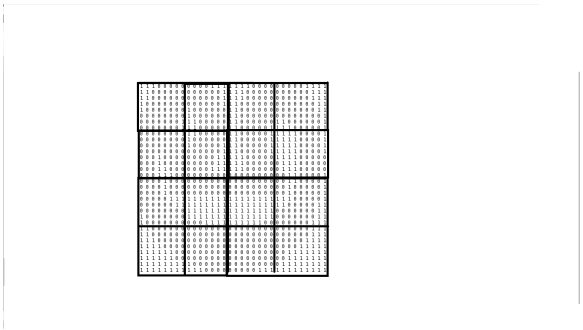


Figure 4.12: Image divided into 16 zones

The minimum distance and misclassified characters of the connected components are shown in the below table

CHARACTER	MINIMUM DISTANCE	CLASSIFIED AS
	943942322728228073819340 8.000000	
	943942302025849795929702 4.000000	
	943942305754760300448317 4.000000	
	943942209048471844002201 6.000000	
	943942305807355553618329 6.000000	
	943942439177961209803571 2.000000	
	943942321268082188117606 4.000000	
	943942310541053502423304 0.000000	
	943942206506212646610344 5.000000	
	943942439177961209803571 2.000000	

Figure 4.13: Characters classified after applying Zoning

But it is found that, the zoning method is not very effective in classification of the characters.

Chapter 5

Conclusion and Future Work

5.1 Conclusion

Based on our experiments it is observed that, features that are derived functions from Gegenbauer moments not seemed to be successful, when the entire image was given as input.It would recognize reliably only some characters.Therefore this attempted method was modified.
Zoning oriented approach was experimented and found results are not so effective.

5.2 Future Work

There are many other polynomials like Hermite polynomials, which have good mathematical properties like orthogonality and separability.
It would be interesting to see how it will work for TELUGU OCR.

Bibliography

[1] Rangachar Kasturi and Lawrence O'Gorman. Document Image Analysis *IEEE Computer Society*, 1754-1759, 1997.

[2] Chakravarthy Bhagvati, Tanuku Ravi, S Mahesh Kumar and Atul Negi. On Developing High Accuracy OCR Systems for Telugu and Other Indian Scripts *Language Engineering Conference*, 2002.

[3] Chakravarthy Bhagvati, Atul Negi and V.V. Suresh Kumar. Non-linear normalization to improve Telugu OCR

[4] Liao, S.; Chiang, A.; Qin Lu; Pawlak, M.. Chinese character recognition via Gegenbauer moments *Pattern Recognition*, 3, 2002.

[5] Hongqing Zhu , Huazhong Shu , Ting Xia , Limin Luo , Jean Louis Coatrieux. Translation and scale invariants of Tchebichef moments. *Pattern Recognition*, 40, 2007.

[6] Peter Kasza. Pseudo-Zernike Moments for Feature Extraction and Chinese Character Recognition. *Pattern Recognition*, 46:2432–2444, 2006.

[7] Chakravarthy Bhagvati, Ravi Kiran Combining Multiple Classifiers for the OCR System, 2004

[8] Liao, S.X., Qin Lu ; A study of moment functions and its use in Chinese character recognition *Proceedings of the Fourth International Conference on Document Analysis and Recognition, 1997*,,1997

[9] A. Negi, C. Bhagvati, B. Krishna "An OCR System for Telugu", ,

Sixth International Conference on Document Analysis and Recognition (ICDAR'01), Vol. 29,2001, pp. 1110-1114

[10] {B.Krishna. Design and implementation of a telugu script recognition system. *Technical report, Department of computer and information sciences, University of Hyderabad.*,

[11] {Ke Han Handwritten Identification and Recognition Using Fuzzy Logic, *Graduate School of Wayne State University, Ph.d Dissertation*, 1995.

[12] P.J.Grother, R.Chellappa and C.L.Wilson Evaluation of pattern classifires for fingerprints and OCR applications. *Pattern Recognition*, 27:i357i365,1994.