EXTRACTION AND SEGMENTATION OF TEXT FROM IMAGE DOCUMENTS

A DISSERTATION

Submitted in partial fulfillment of the requirements for the award of the degree
of
MASTER OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING

By

VIJAY KUMAR

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE -247 667 (INDIA)
JUNE, 2010

Candidate Declaration

I hereby declare that the work being presented in the dissertation report titled **"Extraction and Segmentation of Text from image documents "** in partial fulfillment of the requirement for the award of the degree of Master of Technology in Computer Science and Engineering, submitted in the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee,.is an authentic record of my own work carried out under the guidance of Dr A.K.Sarje, in the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee. I have not submitted the matter embodied in this dissertation report for the award of any other degree.

Dated: 25/06/2010

Place: IIT Roorkee.

Vijay Kumar

(Vijay Kumar)

Certificate

This is to certify that above statements made by the candidate are correct to the best of my knowledge and belief.

25/6/10

Dated:

Place: IIT Roorkee.

Dr. A.K.Sarje,
Professor,
Department of Electronics
and Computer Engineering,
IIT Roorkee, Roorkee,
247667 (India)

ACKNOWLEDGMENTS

First of all and foremost, I would like to thank the Almighty, without whose grace I would not even here. I would like to express my deep sense of gratitude and indebtedness to my guide Dr. A.K.Sarje, for his invaluable guidance and constant encouragement throughout the dissertation. His zeal for getting the best out of his students helped me to perform above my par. I was able to complete this dissertation in time due to the constant motivation and support received from him.

I am also grateful to Dr. R.C. Joshi sir for helping me to clarify some basic and important concepts explored in this dissertation work. I would want to express thanks to my colleagues, Pankaj Sengar and Ashish Kumar for their "taken for granted" help with trivial matters, without which I am sure my work might have hit a dead end.

Last but not the least I would like to thank my family for their constant support, motivation and showing interest which sometimes lead to ray of hope in darkness.

Vijay kumar

(Vijay Kumar)

ABSTRACT

Document images are often obtained by digitizing paper documents like books or manuscripts. Document image analysis systems are becoming increasingly visible in everyday life. Accuracy of any Optical Character Recognition (OCR) heavily depends upon Text segmentation from image document and segmentation of text into line, word, and character.

In this Dissertation we have studied and proposed a new method for text segmentation from image document using Daubechies wavelet and 2-mean classification. For morphology, we have used morphology operation like dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels from object boundaries. We have obtained good accuracy compared to other methods of text segmentation like haar wavelets, Naive Bayes Classifier method and decision tree method. We have used same input image for the above methods and illustrated the corresponding output images. The proposed method for text segmentation from image document has been implemented in MATLAB.

We have also studied and modified the proposed algorithm for segmentation of text into lines, words and characters for Devanagari and Gurmukhi scripts in which we have described the line, word, character and top character segmentation for printed Hindi text in Devanagari script. We have also described the line and word segmentation for printed text in Gurmukhi script. Performance increases in various levels have been obtained. We have observed the performance of segmentation with the help of five documents in devanagari script and five documents in gurmukhi script.

Table of Contents

List of Figures

List of Table

List of Publications

1. Vijay Kumar “Segmentation of Printed Text in Devanagari Script and Gurmukhi Script”, International Journal of Computer Applications (0975 – 8887), Volume 3 – No.8, June 2010.

2. Vijay Kumar, Anil K. Sarje, “Text Segmentation from Image Document”, International Journal of Computer Applications, June 2010. (Under Review)

Chapter 1

INTRODUCTION

1.1 Introduction

Text/Graphics segmentation is a classical problem in Document Image Analysis and has been reported by many researchers. However, until today there is no efficient method for detecting all type of graphics and texts in any orientation from real life documents. In this dissertation work, we are mainly focusing on the Extraction of text from an image document page with the help of Daubechies wavelet and 2-mean classification, and after that segmentation of text into line, words, and character for Devanagari script and also segmentation of text into line and words for Gurmukhi script.

We had progressed in two phase, In First phase, work done on Extraction of text from an image document with the help of Daubechies wavelet and 2-mean classification. In Second phase, work done on segmentation of text into line, words, and character for Devanagari script and also segmentation of text into line and words for Gurmukhi script.

1.2 Motivation

In the face of the very important mass of information exchanged between different organizations, the need for systems allowing the recognition, the indexation, the information retrieval and the automatic classification of complex multi-lingual and multi-script document images has grown continuously. However, most works of backward-conversion of printed document images are limited to textual block recognition without handling complex documents such as letters of information, forms, all types of application sheet, etc. In practice, these documents can be noised, skewed, deformed, multi-lingual, multi-script with irregular textures and may contain several heterogeneous blocks such as annotations, machine print and/or handwritten script, graphics, pictures, logos, photographs, tabular structures. This situation makes it difficult to analyze and recognize document images.

1.3 Problem Statement

In this dissertation work we have proposed and implement a method for Extraction of text from an image documents with the help of Daubechies wavelet and 2-mean classification. After that segmentation of text into line, word and character for Devanagari script and also segmentation of text into line and word for Gurmukhi script.

1.4 Organization of the Report

The report is divided into five chapters including this introductory chapter. The rest of this dissertation report is organized as follows.

Chapter 2

We have discussed the background and literature survey for segmentation of image document into text and picture, after that for segmentation of text into line, word and character for Devanagari script and Gurmukhi script.

Chapter 3

We have discussed and implement the Extraction of text from an image document with the help of Daubechies wavelet and 2-mean classification.

Chapter 4

We have discussed and implement the segmentation of text into line, word and character for Devanagari script, and also segmentation of text into line and word for Gurmukhi script.

Chapter 5

Concludes the dissertation work and gives suggestions for future work.

Chapter 2

BACKGROUND AND LITERATURE SURVEY

2.1 Text Segmentation from Image Document

There are various methods and techniques for text and picture segmentation in image document. In this section different methods and algorithms for text and picture segmentation are studied.

The main text segmentation methods in the literature can be classified into connected component-based, edge-based [2, 3] and texture-based methods [1, 3, and 4]. They are relatively independent of changes in text size and orientation, but having difficulties with complex images with non-uniform backgrounds, for example, if a text string touches a graphical object in the original image, they may form one connected component in the resultant binary image.

The basic idea behind the edge-based algorithms is that the edges of text symbols are typically stronger than those of noise. Edge-based methods are fast and can detect text in complex backgrounds but are restrictive to detect only horizontally or vertically aligned text strings.

In texture-based methods the input image is usually considered as a composite of two (text and non-text) or three (text, picture and background) texture classes. Many segmentation algorithms employ a classification window (block) of a certain size in the hope that all or majority of pixels in the window belong to the same class [5]. Thereafter, a classification algorithm can be used to label each window in the feature space. For example, in [6] the number of classes is two, and a 2-means classification is used to classify each block of the image as text or non-text according to its local energy in the wavelet transform domain. By using a 3-means clustering in each image pixel is labeled as text, picture or background according to a 9-D feature vector based on Gaussian filtering. A large number of statistical and geometrical features have been proposed for texture segmentation. The wavelet transform has become a very effective tool in texture segmentation and classification due to its multi-resolution properties therefore wavelet based features are matter of interest. It provides a powerful transform domain for modeling images that are well characterized by their edges.

The literature on text segmentation is broad in scope but there appears to be very little literature on using machine learning techniques on this subject. Text segmentation algorithm should have adaptation and learning capability, but a learner usually needs much time and training data to achieve satisfactory results, which restricts its practicality. To overcome these problems, M. M. Haji, S. D. Katebi [7] give a simple procedure for generating training data from manually segmented images, then applying a Naive Bayes Classifier (NBC), which is fast both in training and application phase. We have done comparison from D-Tree and Haar wavelet methods [9] with our method. Our method is based on Daubechies wavelet and 2-mean classification .Our method is accurate from existing D-Tree method [8], Naive Bayes Classifier method [7], Haar wavelet method [9]. We have illustrated accuracy comparison from these methods [7, 8, and 9] with our method in implementation section.

2.2 Segmentation of Text

Many methods and algorithms are used for segmentation of text into line, word and character. In this section we have studied different methods and algorithms for segmentation of document into line, word and character.

A lot of research is done in the past on line segmentation of handwritten and printed text. A wide variety of line segmentation methods for handwritten and printed documents are reported in the literature. The various existing methods for line segmentation are categorized as projection based [10, 11], Hough transform based [12], smearing [13], grouping [14], graph based [15], CTM (Cut text Minimum) approach [16], block covering [17] and linear programming.

Segmentation is one of the major stages of character recognition. Recognition of text heavily depends on proper segmentation of text into lines, words and then individual characters or sub-characters for feature extraction and classification of these characters. An error in segmentation may lead to wrong recognition of text and the system may be rendered useless. Segmentation of handwritten words in Devanagari script is a challenging task because of the structural properties of Devanagari character set and writing styles of individuals. Considerable amount of work has been carried out to segment words of machine printed Roman script and there are varied and some well developed techniques. There are references available for segmentation of handwritten text has been done in Roman script also, as shown in [18, 19, and 20] . But very little work has been carried out for Indic scripts like Devnagari, Bengali, Gurmukhi etc. Only few papers are available for segmentation of handwritten and machine printed Devnagari [21, 22, 23, 24 and 25].

Chapter 3
EXTRACTION OF TEXT FROM IMAGE DOCUMENT

3.1 Document Image Analysis

Document image analysis (DIA) is the subfield of digital image processing that aims converting document images to symbolic form for modification, storage, retrieval, reuse and transmission. The objective of document image analysis is to recognize the text and graphics components in images of documents. Document image analysis is the process that performs the overall interpretation of document images. In practice then, a document analysis system performs the basic tasks of image segmentation, layout understanding, symbol recognition and application of contextual rules in an integrated manner.

Image segmentation is useful in high-resolution image analysis. It provides a partitioning of the image into isolated regions, each one representing a different image. The goal of Document Image Analysis systems is the description of the document images expressing the relationships between document components that are meaningful for a reader. DIA cannot be presented as a set of generic methods that could be applied to any documents for any purposes. Before starting the analysis, pre-processing is needed to improve the image quality and to facilitate the further treatments. Physical segmentation of the image into homogeneous parts, spatially organized, is obtained. This segmentation enables to adapt further processes to each specific media (text, graphics).

3.2 Text Segmentation from image document

Documents in which text is embedded in picture are increasingly common today, for example, in magazines, advertisements and web pages. Detection of text from these documents is a challenging problem. Text extraction has a vast number of applications:

- Text searches in Images - Currently, Image searches deliver inaccurate results as they do not search the image content. Text extraction would enable better searching by extracting the content of an image.

- Content based Indexing - For the purpose of archiving and indexing documents, the content of the document is required in the digital format. Knowledge about the text content of documents can help in the building of an intelligent system which archives and indexes the printed documents.
- Reading foreign language text - One of the common problems faced by a person in foreign land is that of communication, understanding road signs, signboards etc. Text segmentation from image document to make easier such problems by reading the text information from the image scenes which are captured by a camera.
- Archiving documents - Archives of paper documents in offices or other printed material like magazines and newspapers can be electronically converted for more efficient storage and instant delivery to home or office computers.

3.2.1 Image Acquisition

The image is then transferred to the PC using either cable or through Bluetooth device. In case of document image recognition, a set of printed and hand written documents are digitized through a scanner by manually placing the document on the bed of the scanner. Image Acquisition is the process of collection of images for text and picture segmentation in image document. We have used scanned image for text and picture segmentation in image document.

3.2.2 Image preprocessing

A technique in which the data from an image are digitized and various mathematical operations are applied to the data, generally with a digital computer, in order to create an enhanced image that is more useful or pleasing to a human observer, or to perform some of the interpretation and recognition tasks usually performed by humans. Also known as picture processing.

The analysis of a picture using image processing that can identify shades, colors and relationships that cannot be perceived by the human eye. Image processing is used to solve identification problems, such as in forensic medicine or in creating weather maps from satellite pictures. It deals with images in bitmapped graphics format that have been scanned in or captured with digital cameras. [27]

The color images are then converted to grey level images by finding the grey value of each pixel located at (i,j) from the 24-bit color value of it using the following formula which shown in reference [20].

$$\text{grey}(i, j) = 0.59\ \text{red}(i, j) + 0.30\ \text{green}(i, j) + 0.11\ \text{blue}(i, j)$$

3.2.3 Daubechies wavelet

The Daubechies wavelets are a family of orthogonal wavelets defining a discrete wavelet transform and characterized by a maximal number of vanishing moments for some given support [28]. The names of the Daubechies family wavelets are written dbN, where N is the order, and db the "surname" of the wavelet.

Any discussion of wavelets begins with Haar wavelet, the first and simplest. Haar wavelet is discontinuous, and resembles a step function. It represents the same wavelet as Daubechies db1.

3.2.4 Block processing

The Block Processing block extracts submatrices of a user-specified size from each input matrix. It sends each submatrix to a subsystem for processing, and then reassembles each subsystem output into the output matrix. Hence repeat user-specified operation on submatrices of input matrix. If we want to divide an image into blocks and process each block individually, we can use the function blkproc. This function will allow for me to process distinct blocks as well as overlapping blocks. With a little creativity, this function can be used to eliminate many loops that would otherwise be necessary. [26] In our implementation, we have used [3 3] sub matrix for block processing.

3.2.5 K-means Clustering

The K-mean algorithm is an iterative technique that is used to partition an image into *K* clusters. The basic algorithm is:

1. Pick *K* cluster centers, either randomly or based on some heuristic.
2. Assign each pixel in the image to the cluster that minimizes the distance between the pixel and the cluster center
3. Re-compute the cluster centers by averaging all of the pixels in the cluster

4. Repeat steps 2 and 3 until convergence is attained (e.g. no pixels change clusters)

The k-means algorithm is an evolutionary algorithm that gains its name from its method of operation. The algorithm clusters observations into k groups, where k is provided as an input parameter. [29]
We have used 2-means classification in our implementation. One group of white pixel and second group of black pixel are used in 2-means classification.

3.2.6 Post Processing

Post processing attempts to increase the quality of a mask image. Post processing has been done with the help of morphology. Morphology is a broad set of image processing operation. Morphological operations apply a structuring element to an input image, creating an output image of the same size. In a morphological operation, the value of each pixel in the output image is based on a comparison of the corresponding pixel in the input image with its neighbors. By choosing the size and shape of the neighborhood, you can construct a morphological operation that is sensitive to specific shapes in the input image.

The most basic morphological operations are dilation and erosion. Dilation adds pixels to the boundaries of objects in an image, while erosion removes pixels on object boundaries. The number of pixels added or removed from the objects in an image depends on the size and shape of the structuring element used to process the image. In the morphological dilation and erosion operations, the state of any given pixel in the output image is determined by applying a rule to the corresponding pixel and its neighbors in the input image. The rule used to process the pixels defines the operation as dilation or erosion. This table lists the rules for both dilation and erosion.[30]

Table 3.1: Rules for Dilation and Erosion

Operation	Rule
Dilation	The value of the output pixel is the maximum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to the value 1, the output pixel is set to 1.
Erosion	The value of the output pixel is the minimum value of all the pixels in the input pixel's neighborhood. In a binary image, if any of the pixels is set to 0, the output pixel is set to 0.

3.2.7 Proposed Algorithm for text segmentation from image document

```
Step1:- Read one image and store in X_in.
Step2:- Image converted from RGB into Gray scale image.
Step3:- Find the no. of rows (nr) and no. of column (nc) in an image X_in.
Step4:- compute the approximation coefficients matrix A and details coefficients matrices H, V, and D (horizontal, vertical, and diagonal, respectively), obtained by Daubechies wavelet decomposition of the input matrix X_in. We get approximation coefficients matrix cD2 due to two level decomposition.
Step5:- Apply fun = inline ('sum (abs(x (:)))'); and used the value of fun in blkproc function.
Step6:- Apply block processing for matrix cD2 by applying the fun to each distinct 3-by-3 block of cD2 and return E1.
Step7:- Apply 2-means classification and return an n-by-1 vector IDX containing the cluster indices of each point in E1. And construct C matrix.
Step8:- if
            Maximum pixel value of first row in C > Maximum pixel value of second row in C
             IDX (IDX= =1) = 1;
             IDX (IDX= =2) = 0;
      else
            IDX (IDX= =1) = 0;
             IDX (IDX= =2) = 1;
Step9:- Change the size of IDX according to the size of E1.
Step10:- Write the image according IDX matrix in tmp.bmp format.
Step11:- Apply morphology for post processing and obtain the mask image L.
Step12:- Convert the mask image into segmented image.
```

3.2.8 Flow work diagram

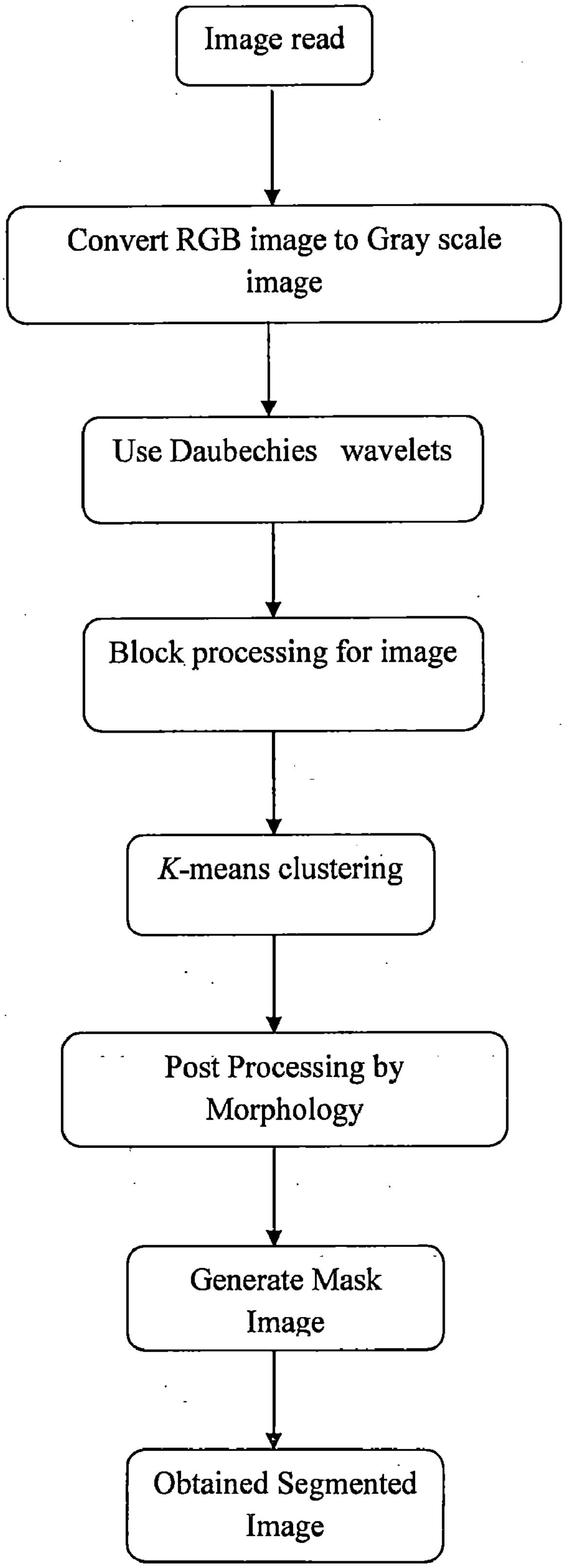
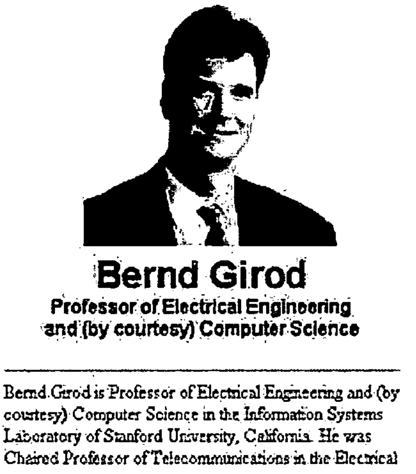


Figure 3.1: Flow work diagram for text segmentation

3.2.9 Experimental Results

We have developed segmentation of text and picture from image document with the help of Daubechies wavelet and 2-mean classification using MATLAB R2009a. Here we have considered some images and illustrate the segmentation of text and picture.



Bernd Girod

Professor of Electrical Engineering and (by courtesy) Computer Science

Bernd Girod is Professor of Electrical Engineering and (by courtesy) Computer Science in the Information Systems Laboratory of Stanford University, California. He was Chaired Professor of Telecommunications in the Electrical

Figure 3.2: Original image

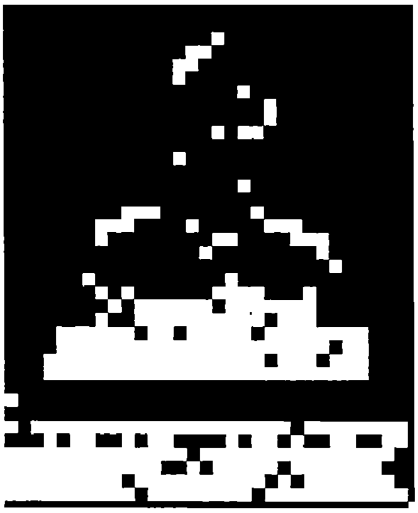


Figure 3.3: Output image before post-processing by Decision Trees method

We have considered input the image Figure 3.2. By Decision Tree segmentation method, the output without post-processing is shown in Figure 3.3 and the output (after post-processing) and final output are shown in Figure 3.4 and Figure 3.5 respectively:

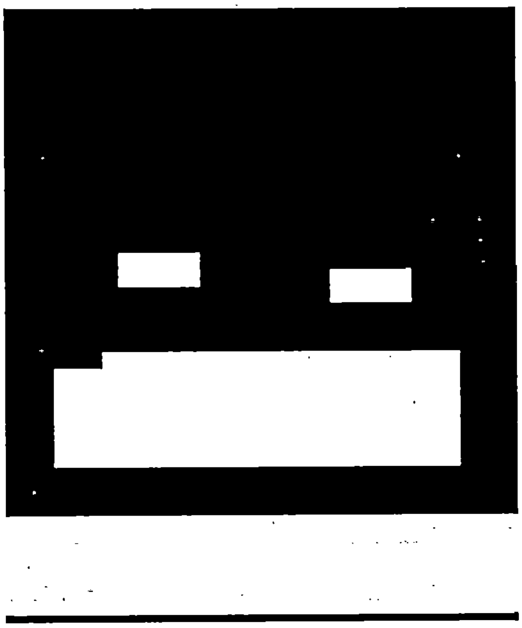
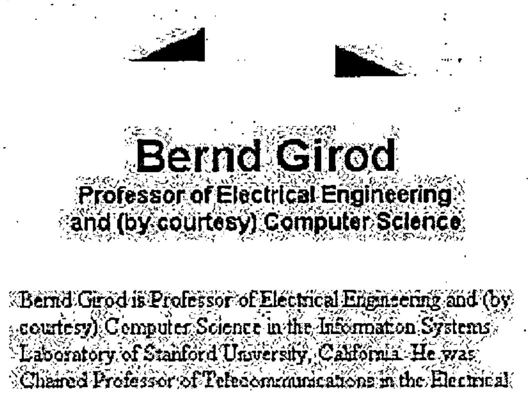


Figure 3.4: Output mask image after post-processing by Decision Trees method



Bernd Girod
Professor of Electrical Engineering
and (by courtesy) Computer Science

Bernd Girod is Professor of Electrical Engineering and (by courtesy) Computer Science in the Information Systems Laboratory of Stanford University, California. He was Chaired Professor of Telecommunications in the Electrical

Figure 3.5: Segmented image by Decision Trees method

Another text segmentation method as described in [9], the method is based on high frequency haar wavelet coefficients. If we give input the same image (Figure 3.2) then following images will be resulted:

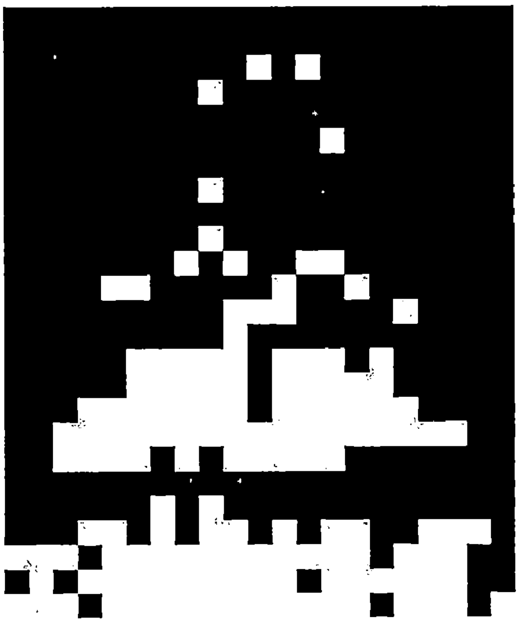


Figure 3.6: Output mask image before post-processing by Haar wavelet

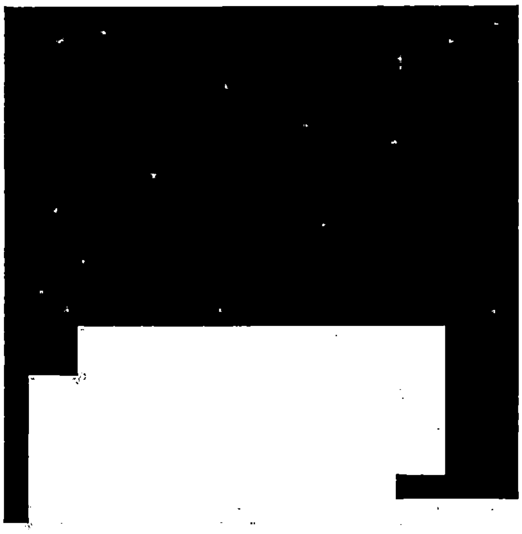
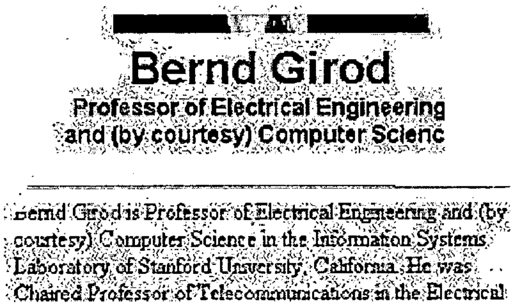


Figure 3.7: Output mask image after post-processing by Haar wavelet



Bernd Girod
Professor of Electrical Engineering
and (by courtesy) Computer Scienc

Bernd Girod is Professor of Electrical Engineering and (by courtesy) Computer Science in the Information Systems Laboratory of Stanford University, California. He was Chaired Professor of Telecommunications in the Electrical

Figure 3.8: Segmented image by Haar wavelet method

Our text segmentation method, the method is based on Daubechies wavelet and 2-mean classification. If we give input the same image (Figure 3.2) then following images will be resulted:

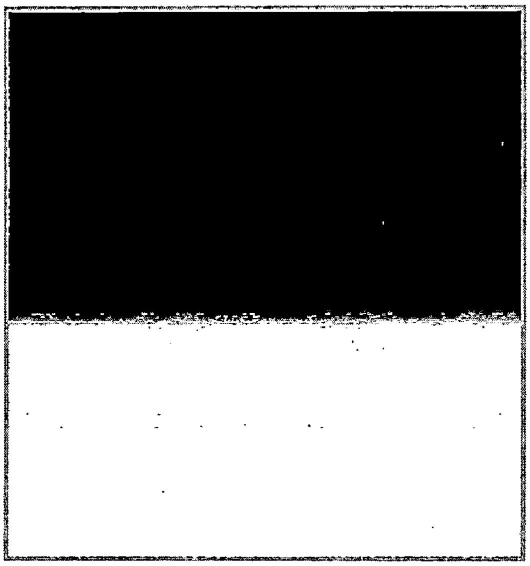
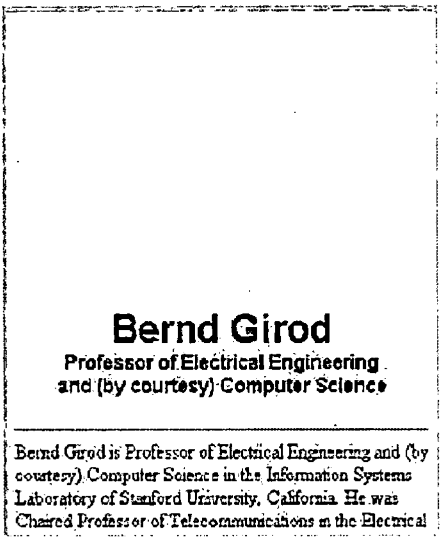


Figure 3.9: Output mask image after post-processing by our method

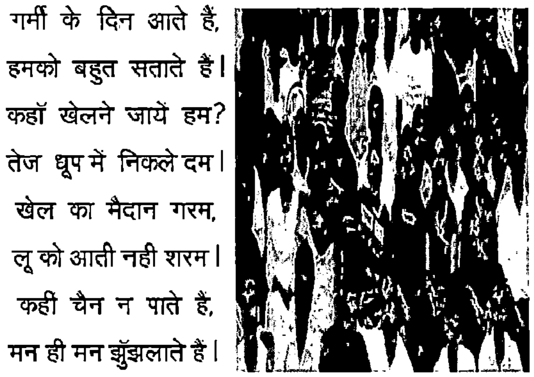


Bernd Girod
Professor of Electrical Engineering
and (by courtesy) Computer Science

Bernd Girod is Professor of Electrical Engineering and (by courtesy) Computer Science in the Information Systems Laboratory of Stanford University, California. He was Chaired Professor of Telecommunications in the Electrical

Figure 3.10: Segmented image by our method

We have done experiment for another image by our method.



गर्मी के दिन आते हैं,
हमको बहुत सताते हैं।
कहाँ खेलने जायें हम?
तेज धूप में निकले दम।
खेल का मैदान गरम,
लू को आती नही शरम।
कहीं चैन न पाते हैं,
मन ही मन झुँझलाते हैं।

Figure 3.11: Original image

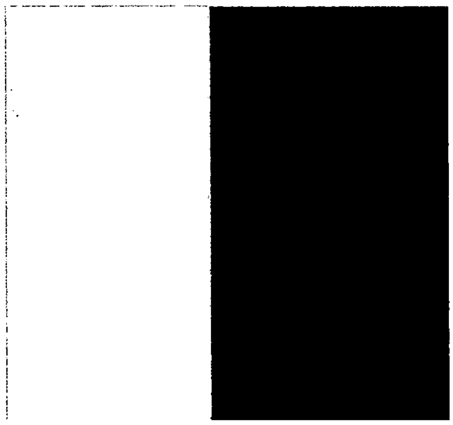
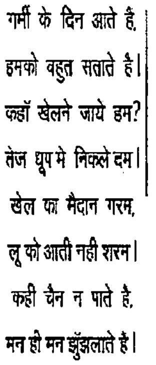


Figure 3.12: Output mask image of figure 3.11



गर्मी के दिन आते है,
हमको वहुत सताते है।
कहाँ खेलने जाये हम?
तेज धूप मे निकले दम।
खेल का मैदान गरम,
लू को आती नही शरम।
कही चैन न पाते है,
मन ही मन झुँझलाते है।

Figure 3.13: Segmented image of figure 3.11

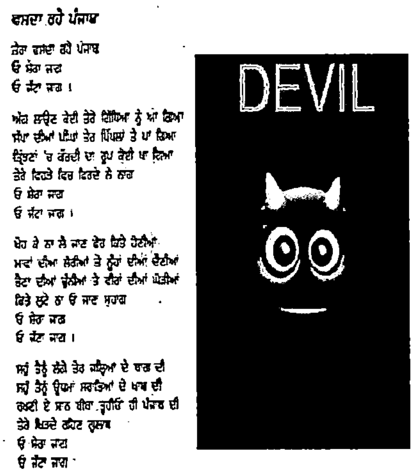


Figure 3.14: Original image

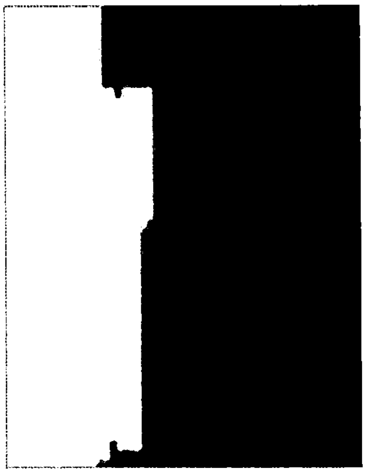


Figure 3.15: Output mask image of Figure 3.14

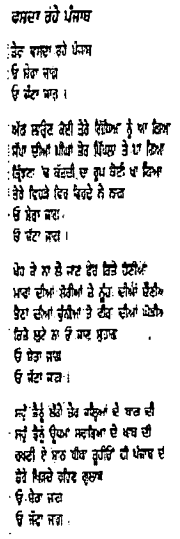
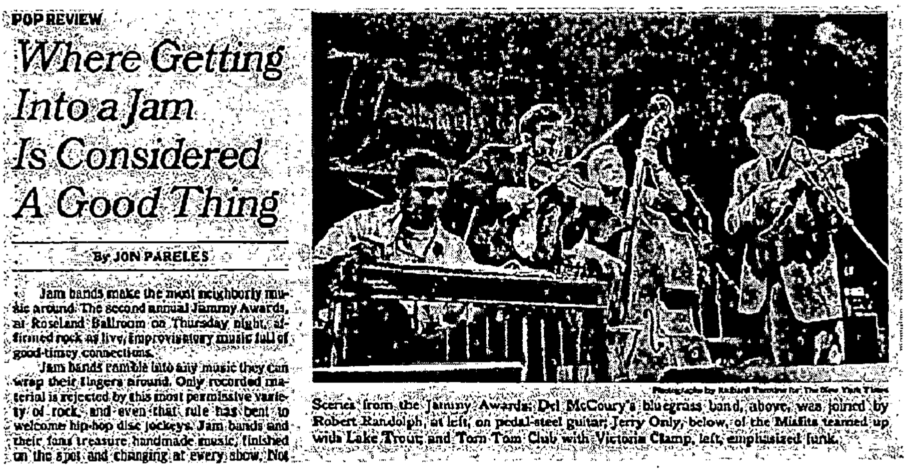


Figure 3.16: Segmented image of Figure 3.14

We have done some experiment and comparison with other method.



POP REVIEW

Where Getting Into a Jam Is Considered A Good Thing

By JON PARELES

Jam bands make the most neighborly music around. The second annual Jammy Awards, at Roseland Ballroom on Thursday night, affirmed rock as live, improvisatory music full of good-timey connections.

Jam bands ramble into any music they can wrap their fingers around. Only recorded material is rejected by this most permissive variety of rock, and even that rule has bent to welcome hip-hop disc jockeys. Jam bands and their fans treasure handmade music, finished on the spot and changing at every show. Not

Photographs by Richard Termine for The New York Times

Scenes from the Jammy Awards: Del McCoury's bluegrass band, above, was joined by Robert Randolph, at left, on pedal-steel guitar; Jerry Only, below, of the Misfits teamed up with Lake Trout; and Tom Tom Club with Victoria Clamp, left, emphasized funk.

Figure 3.17: Original image

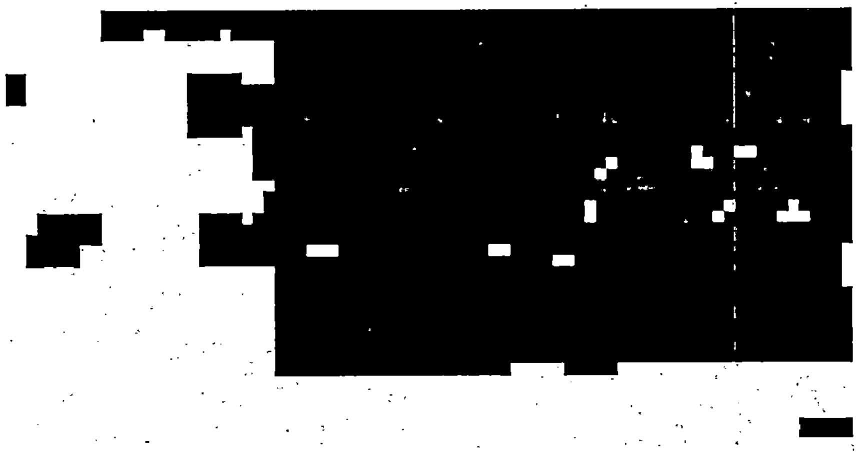
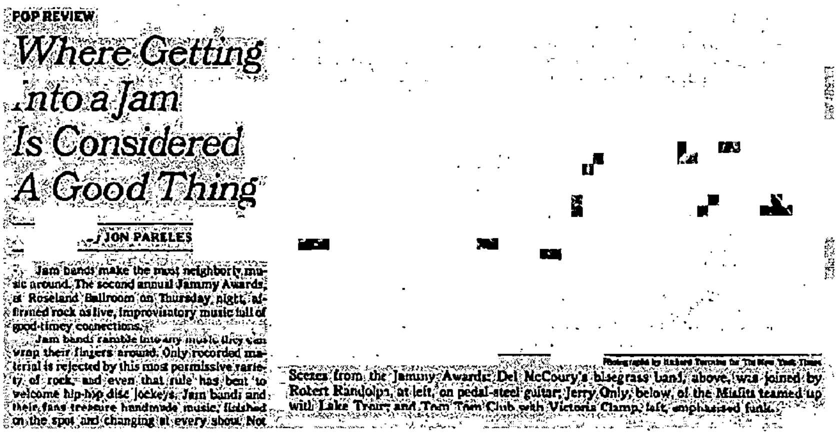


Figure 3.18: mask image after post processing by Naive Bayes Classifier method



POP REVIEW

Where Getting Into a Jam Is Considered A Good Thing

By JON PARELES

Jam bands make the most neighborly music around. The second annual Jammy Awards, at Roseland Ballroom on Thursday night, affirmed rock as live, improvisatory music full of good-timey connections.

Jam bands ramble into any music they can wrap their fingers around. Only recorded material is rejected by this most permissive variety of rock, and even that rule has bent to welcome hip-hop disc jockeys. Jam bands and their fans treasure handmade music, finished on the spot and changing at every show. Not

Photographs by Richard Termine for The New York Times

Scenes from the Jammy Awards: Del McCoury's bluegrass band, above, was joined by Robert Randolph, at left, on pedal-steel guitar; Jerry Only, below, of the Misfits teamed up with Lake Trout; and Tom Tom Club with Victoria Clamp, left, emphasized funk.

Figure 3.19: Segmented image of Figure 3.17 by Naive Bayes Classifier method

Mask image and segmented image according our method is illustrate in Figure 3.20 and Figure 3.21 respectively for the same image given in Figure 3.17.

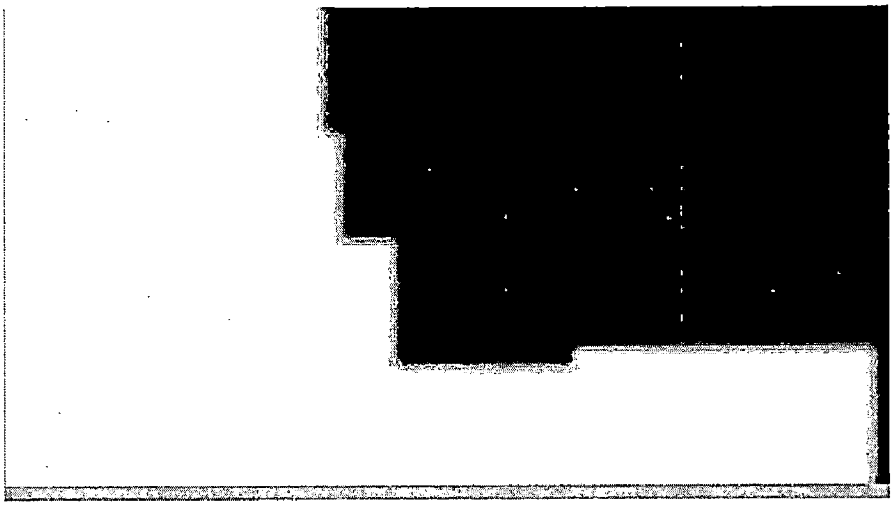
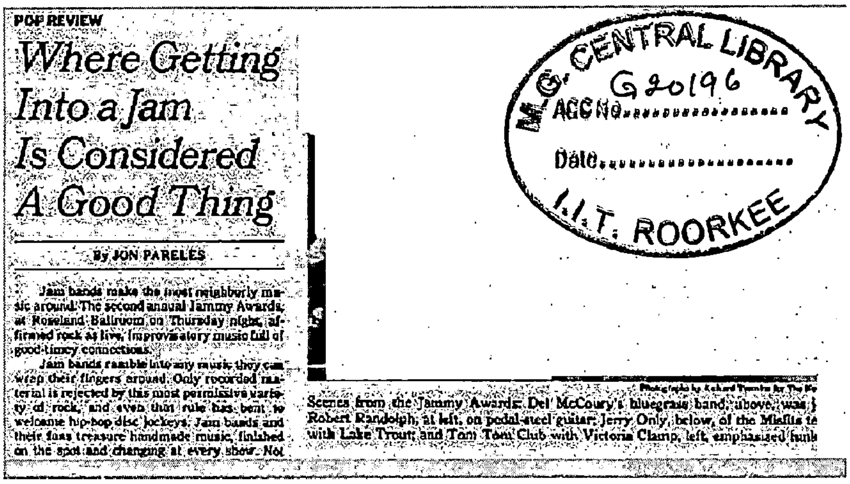


Figure 3.20: Output mask image of Figure 3.17 by our method



POP REVIEW

Where Getting Into a Jam Is Considered A Good Thing

By JON PARELES

Jam bands make the most neighborly music around. The second annual Jammy Awards, at Roseland Ballroom on Thursday night, affirmed rock as live, improvisatory music full of good-timey connections.

Jam bands ramble into any music they can wrap their fingers around. Only recorded material is rejected by this most permissive variety of rock, and even that rule has bent to welcome hip-hop disc jockeys. Jam bands and their fans treasure handmade music, finished on the spot and changing at every show. Not

Photographs by Richard Termine for The Ne

Scenes from the Jammy Awards: Del McCoury's bluegrass band, above, was
Robert Randolph, at left, on pedal-steel guitar; Jerry Only, below, of the Misfits te
with Lake Trout; and Tom Tom Club with Victoria Clamp, left, emphasized funk

Figure 3.21: Segmented image of Figure 3.17 by our method

We have illustrated results for segmentation accuracy of original image (in Figure 3.2 and Figure 3.17) in Figure 3.22 and Figure 3.23 respectively. For original image (Figure 3.2), we have obtained segmentation accuracy 99.894%. And for original image (in Figure 3.17), we have obtained segmentation accuracy 99.60%. Hence our method is easier and accurate from D-Tree method (Accuracy 99.392%), Haar wavelet method (98.39%), and Naïve Bayes Classifier method (Accuracy 99.5%).

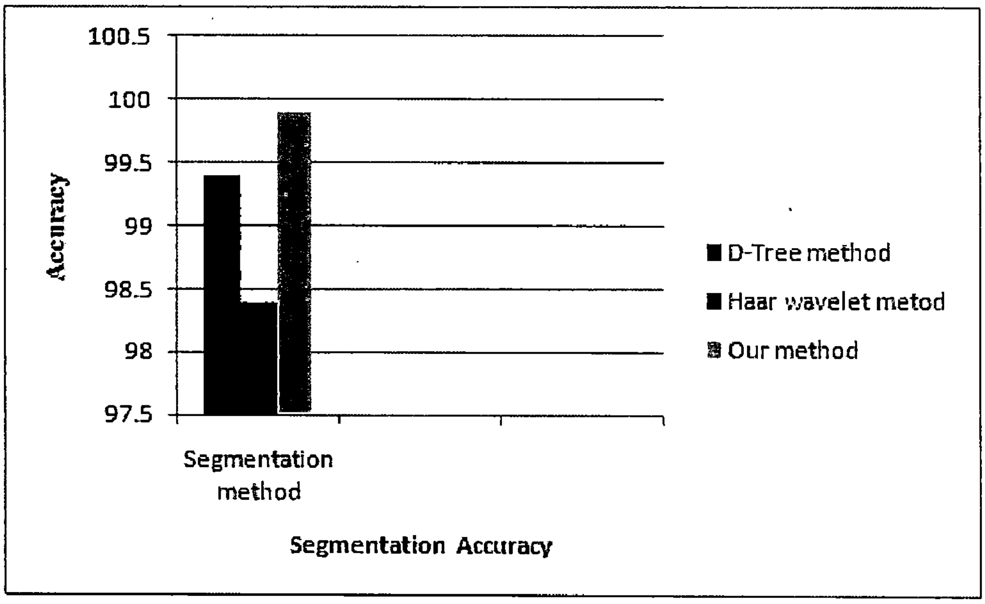


Figure 3.22: Segmentation accuracy for original image given in Figure 3.2

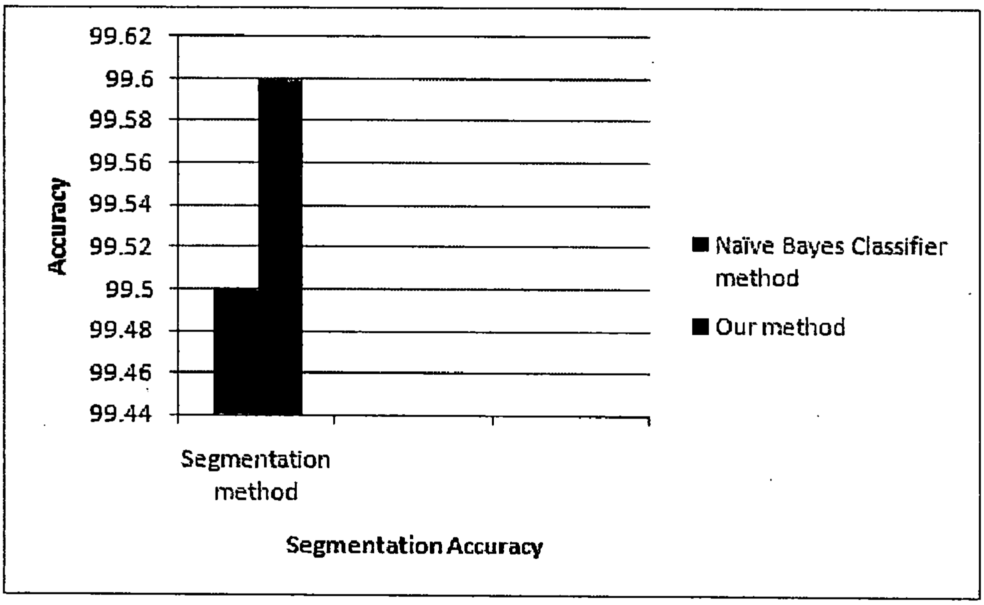


Figure 3.23: Segmentation accuracy for original image given in Figure 3.17

Chapter 4

SEGMENTATION OF TEXT INTO LINE, WORD, AND CHARACTER

In this chapter, we have modified algorithm [31] and proposed the algorithm for segmentation of text into line, word and character for Devanagari script and Gurmukhi script.

Segmentation refers to the process of partitioning a digital image into multiple segments (sets of pixels). The goal of segmentation is to simplify and change the representation of an image into something that is more meaningful and easier to analyze. Text segmentation is a process in which the text image is segregated into units of patterns that seem to form characters. All recognition algorithms depend on the segmentation algorithm to break up the image into individual characters. Segmentation process involves three steps namely line segmentation, word segmentation and character segmentation.

Line segmentation is the process in which from the image, we extract only lines or differentiate the lines.

Word segmentation is the process in which from the line segmentation we extract only words. As we know that there is a distance between one word to another word this concept is used for word segmentation. Word segmentation is the problem of dividing a string of written language into its component words.

Character segmentation is the process in which from the word segmentation we extract only characters. Character segmentation is a crucial step of OCR systems as it extracts meaningful regions for analysis. This step attempts to decompose the image into classifiable units called character.

4.1 Characteristics of Devanagari Script

Devanagari is used in many Indian languages like Hindi, Nepali, Marathi, Sindhi etc. More than 300 million people around the world use Devanagari script. This script forms the foundation of Indian languages. So Devanagari script plays a very major role in the development of literature and manuscripts. Devanagari script has about 11 vowels and 33 consonants. And Devanagari word is written into the three strips namely: a core strip, a top strip, and a bottom strip as shown in figure 4.1. The core strip and top strip are differentiated by the header, while the lower modifier is attached to the core character.

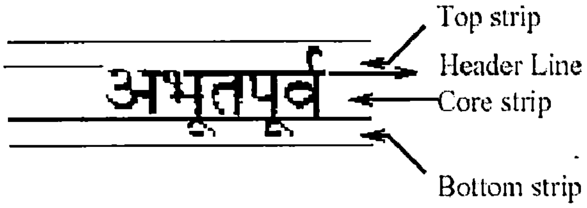


Figure 4.1: Three strips of Devanagari word

OCR for Devnagari script becomes even more difficult when compound character and modifier characteristics are combined in 'noisy' situations. The image below illustrates a Devanagari document with background noise in Figure 4.2. We can clearly see that compound characters and modifiers are difficult to detect in this image because the image background is not uniform in color, and marks are present that must be distinguished from characters.

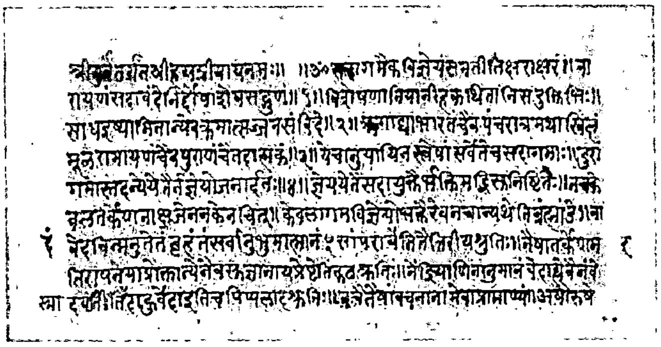

[illegible]

Figure 4.2: Image with background noise

4. 2 Characteristics of Gurmukhi Script

Gurmukhi script alphabet consists of 41 consonants and 12 vowels [12]. Some characters in the form of half characters are present in the feet of characters. Writing style is from left to right. In Gurmukhi, There is no concept of upper or lowercase characters. A line of Gurmukhi script can be partitioned into three horizontal zones namely, upper zone, middle zone and lower zone. Consonants are generally present in the middle zone. These zones are shown in Figure 4.3. The upper and lower zones may contain parts of vowel modifiers and diacritical markers. [13]

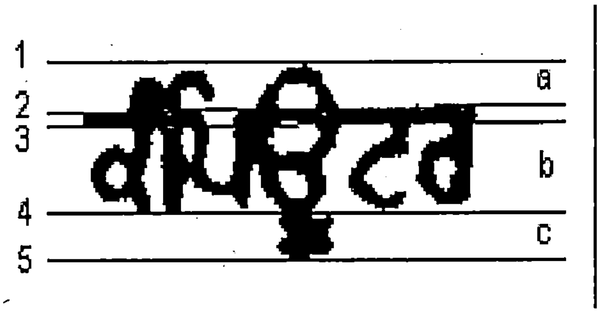


Figure 4.3: a) Upper zone from line number 1 to 2, b) Middle Zone from line number 3 to 4, c) lower zone from line number 4 to 5.

4.3 Image Categories and Pre –Processing

The various categories of the images that could be fed as an input which is in three categories. Binary level images, pseudo color and true color images. For a binary level image, the preprocessing required is minimal. There are two colors, a foreground and a background color. The text is usually represented in the foreground. So we would need to look for the foreground components and perform the analysis.

The next category of images is the pseudo color images. The best example of the pseudo color images are GIFs. It makes use of only 256 different colors. The True color images make use of 16M colors. In this report we use GIF images, TIFF image, JPEG image and JPG images.

4.4 Segmentation For Devanagari Script And Gurmukhi Script

Segmentation is a classifier which helps to fragment each character from a word present in a given image / page. The objective of the segmentation is to extract each character from the text present in the image. Here we have considered bottom strip with core strip. The process of segmentation *mainly* follows the following pattern:

- First, it identifies the page layout
- After that, it identifies the line from the page
- Identifies the word from that line, and
- Finally, identifies the character from that word.

4.5 Proposed Algorithm for Segmentation of Devanagari Script and Gurmukhi Script

Step 1: Line Segmentation

In line segmentation our aim is to draw of one upper horizontal line and one lower horizontal line for each line of text image. The steps for line segmentation are as follow:

- Horizontal scanning for the input image
- Using the Horizontal scanning, find the points from which the line starts and ends.
- For a line of text, upper line is drawn at a point where we start finding black pixels and lower line is drawn where we start finding absence of black pixels. And the process continues for next line and so on.

Step 2: Word Segmentation

- Vertical scanning for each segmented line.
- Using the vertical scanning, find the points from which the word starts and ends.
- Vertical lines are drawn at starting and ending points for each word.

Step 3: Character Segmentation

- Horizontal scanning for each segmented line.
- From the horizontal scanning, find the row which consists of maximum value.
- The row which consists of maximum value of black pixel for each line is actually the row which consists of Header line.
- Using the vertical scanning for each segmented word in below of header line.
- Using the vertical scanning for each segmented word in above of header line.
- Using the vertical scanning, find the points from which the character starts and ends.
- Draw line according these coordinate.

4.6 Packages Used

import java.awt.*;// this package is a abstract window toolkit for applets design for interaction with user.

import java.awt.event.*; //This package is supporting handled event are those generated by mouse, keyboard and other control such as push button etc

import javax.swing.*; //swing is a set of class that provide a more powerful and flexible component than in AWT.

import javax.swing.JOptionPane; //It is a subpackage of swing class which contain option panel.

import java.io.*;//This package is used for INPUT from user and OUTPUT by program or console stream

import java.util.*; //This package contain some of the most exciting enhancement like : collection and contain a wide assortment of classes and interface that support broad range of functionality.

import java.awt.image.*; //This package use to support graphic images pictures.

4.7Designing Panel Frame Buttons and Scrollbars

```
//... create Button and its listeners

    JButton openButton = new JButton("Open");

    JButton lineButton = new JButton("line segment");

    JButton wordButton=new JButton("word segment");

    JButton charButton=new JButton("char segment");

    JButton clearButton=new JButton("clear");


//setting tool tips for various buttons

    openButton.setToolTipText("click here to choose a file");

    lineButton.setToolTipText("click here for line segmentation");

    wordButton.setToolTipText("click here for word segmentation");

    charButton.setToolTipText("click here for char segmentation");

    clearButton.setToolTipText("click here to clear the panel");
```

```
//adding mouse listener to various buttons

    openButton.addActionListener(new OpenAction());

    lineButton.addActionListener(new LineAction());

    wordButton.addActionListener(new wordAction());

    charButton.addActionListener(new charAction());

    clearButton.addActionListener(new clearAction());



//... Create contant pane, layout components

    JPanel content = new JPanel();

    JMenuBar bar=new JMenuBar();

    setJMenuBar(bar);

    JMenu helpmenu=new JMenu("Help");

    helpmenu.setMnemonic('H');

    JMenuItem aboutopen=new JMenuItem("About open");

    JMenuItem lineseg=new JMenuItem("Line segmentation");

// Create JPanel canvas to hold the picture

    imagepanel = new DrawingPanel();

// Create JScrollPane to hold the canvas containing the picture

    JScrollPane scroller = new JScrollPane (

            JScrollPane.VERTICAL_SCROLLBAR_ALWAYS,

                    JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
```

```
      scroller.setPreferredSize(new Dimension(500,300));
      scroller.setViewportView(imagepanel);
      scroller.setViewportBorder(
        BorderFactory.createLineBorder(Color.black));
// Add scroller pane to Panel
      Content. add (scroller,"Center");
// Set window characteristics
    this.setTitle("File Browse and View");
    this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.setContentPane(content);
    this.pack();
```

4.8 Important Methods

```
public int wordseg(int lineno, int w, int h, int vHisto[])
//this above method is used for word by word segmentation
public int lineseg(int w, int h, int hHisto[])
//this above method is used for Line by Line segmentation Horizontally
public int hline(int ln, int wn, int w, int h, int hHisto[])
//this above method is used for Line by Line selection Horizontally
public void ccharseg(int ln, int wn, int w, int h, int vHisto[])
//this above method is used for vertically selecting single character segmentation
```

```
public boolean accept (File f)

// this function is internally used for the Filtering action

public String getDescription ()

// this function is internally used for the Filter Option drop down menu
```

4.9 Development Requirements

4.9.1 Software Requirements

During the solution development the following softwares were used:

- Microsoft Visual Studio
- JDK1.4
- Swings
- JCreator

4.9.2 Hardware Requirements

During the solution development the following hardaware specifications were used:

- 2.0 GHZ Core2Duo Processor
- Minimum 1GB Ram

4.9.3 Input Requirements

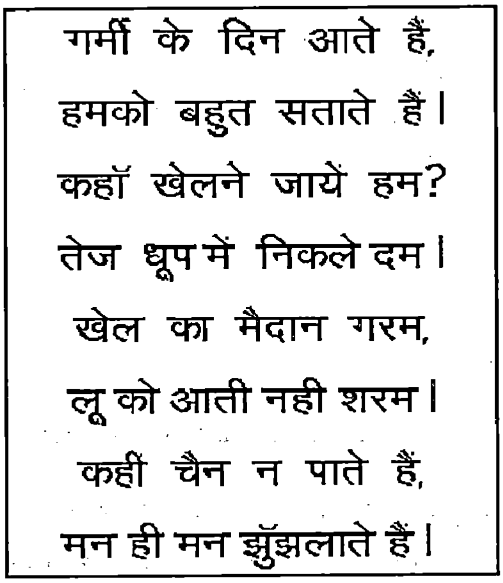
Scanned image as the input.

4.10 Technologies Utilized

Here we used Swing Technology. Swing is a GUI toolkit for Java. Swing is one part of the Java Foundation Classes (JFC). Swing includes graphical user interface (GUI) widgets such as text boxes, buttons, split-panes, and tables. Swing widgets provide more sophisticated GUI components than the earlier Abstract Windowing Toolkit. Since they are written in pure Java, they run the same on all platforms, unlike the AWT which is tied to the underlying platform's windowing system. Swing supports pluggable look and feel– not by using the native platform's facilities, but by roughly emulating them. This means we can get any supported look and feel on any platform. The disadvantage of lightweight components is possibly slower execution. The advantage is uniform behavior on all platforms.

4.11 Experimental Results And Discussions

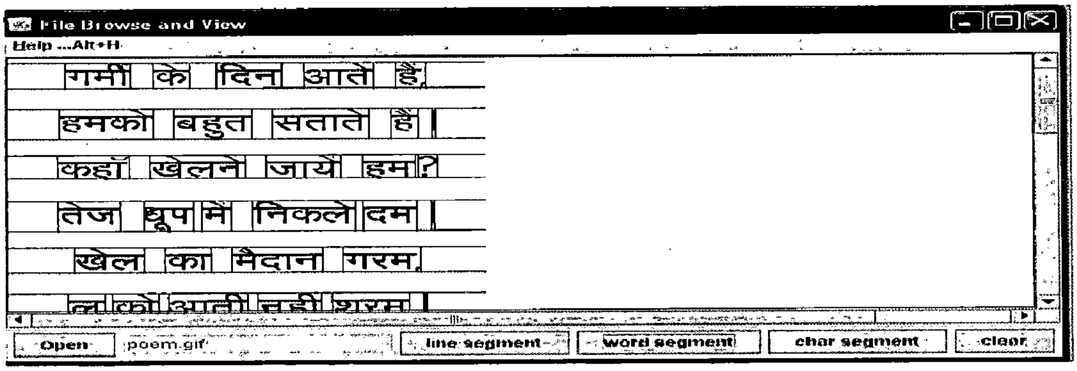
We have collected 10 printed documents, which is document-1 to document-5 in Devanagari Script and from document-6 to document-10 in Gurmukhi Script. We have shown document-3 in figure 4.4 and we have illustrated the number of lines and each line contains the number of words, characters and top characters with the help of Table-4.1.



गर्मी के दिन आते हैं,
हमको बहुत सताते हैं।
कहॉ खेलने जायें हम?
तेज धूप में निकले दम।
खेल का मैदान गरम,
लू को आती नही शरम।
कहीं चैन न पाते हैं,
मन ही मन झुंझलाते हैं।

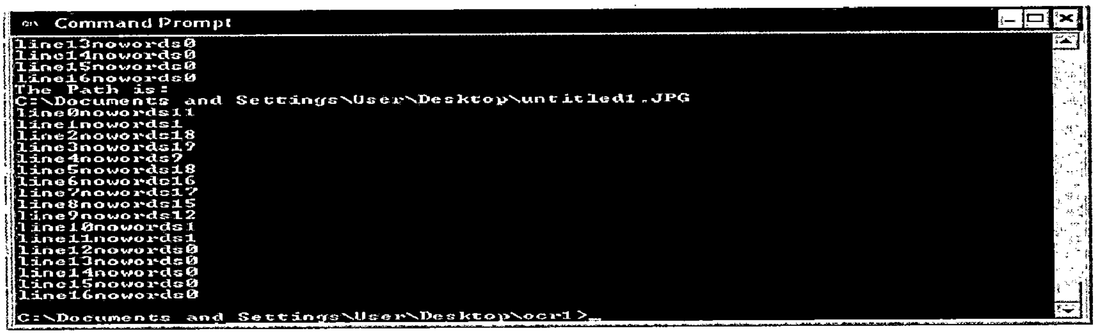
Figure 4.4: Document-3 in Devanagari Script.

We have constructed the Table-4.1, which shows the accuracy of word, character and top character segmentation for document-3 in Devanagari script by Figure 4.4. Which is also illustrate the recognize words, characters, and top characters with respect of original words, original characters and original top characters respectively for each line of document-3. And we have illustrated line and word segmentation of document 3 in Figure 4.5. Output of Line and Word segmentation of document- 3 shown in Figure 4.6 which is illustrated that how many number of words in each line. For Character segmentation of document-3, we have pushed the button 'character segment' then output will be shown in Figure 4.7 by which we have illustrated the character segmentation and top character segmentation. And in Figure 4.8, we have illustrated the output of Character segmentation for document-3 by which illustrated number of character and top character in each line of document-3.



गर्मी के दिन आते हैं,
हमको बहुत सताते हैं।
कहाँ खेलने जायें हम?
तेज धूप में निकले दम।
खेल का मैदान गरम,

Figure 4.5: Line and Word segmentation of document 3



```
Command Prompt
line13nowords0
line14nowords0
line15nowords0
line16nowords0
The Path is:
C:\Documents and Settings\User\Desktop\untitled1.JPG
line0nowords11
line1nowords1
line2nowords18
line3nowords17
line4nowords7
line5nowords18
line6nowords16
line7nowords17
line8nowords15
line9nowords12
line10nowords1
line11nowords1
line12nowords0
line13nowords0
line14nowords0
line15nowords0
line16nowords0

C:\Documents and Settings\User\Desktop\ocr1>_
```

Figure 4.6: Output of Line and Word segmentation for document 3

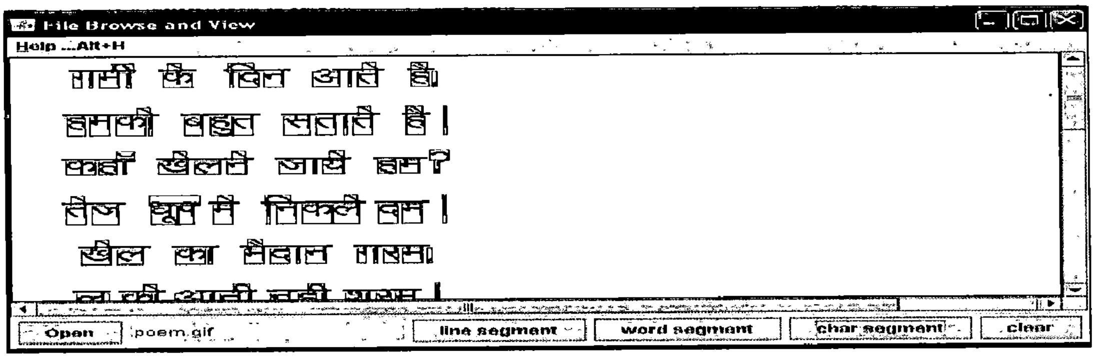
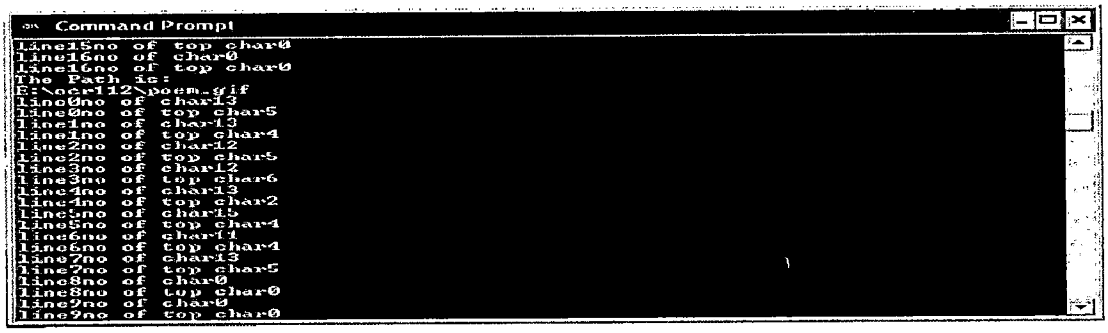


Figure 4.7: Character segmentation of document 3

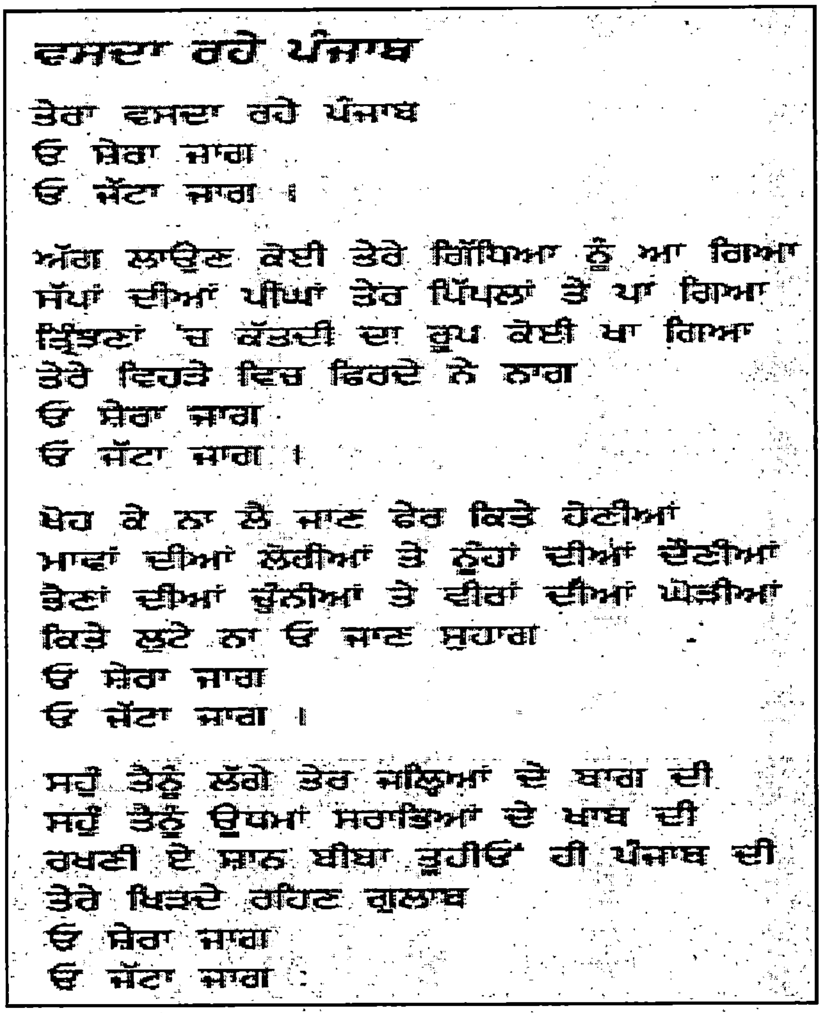


```
Command Prompt
line15no of top char0
line16no of char0
line16no of top char0
The Path is:
E:\ocr112\poem.gif
line0no of char13
line0no of top char5
line1no of char13
line1no of top char4
line2no of char12
line2no of top char5
line3no of char12
line3no of top char6
line4no of char13
line4no of top char2
line5no of char15
line5no of top char4
line6no of char11
line6no of top char4
line7no of char13
line7no of top char5
line8no of char0
line8no of top char0
line9no of char0
line9no of top char0
```

Figure 4.8: Output of Character segmentation for document 3

Table 4.1:-Result of word, Character and top character Segmentation of Document-3 in Devanagri script

Line no.	No. of words		No. of characters		No. of top characters	
	Original words	Recognize words	Original char	Recognize characters	Original top characters	Recognize top characters
0	5	5	13	12	5	5
1	5	5	13	13	4	4
2	5	5	12	12	5	5
3	6	6	12	12	6	5
4	4	4	13	13	2	2
5	6	6	15	15	4	4
6	5	5	11	11	4	4
7	6	6	13	13	5	5
Total	42	42	102	101	35	34
Accuracy	100%		99%		97%	



ਵਸਦਾ ਰਹੇ ਪੰਜਾਬ

ਤੇਰਾ ਵਸਦਾ ਰਹੇ ਪੰਜਾਬ
ਓ ਸ਼ੇਰਾ ਜਾਗ
ਓ ਜੱਟਾ ਜਾਗ ।

ਅੱਗ ਲਾਉਣ ਕੋਈ ਤੇਰੇ ਗਿੱਧਿਆਂ ਨੂੰ ਆ ਗਿਆ
ਸੱਪਾਂ ਦੀਆਂ ਪੀਂਘਾਂ ਤੇਰ ਪਿੱਪਲਾਂ ਤੇ ਪਾ ਗਿਆ
ਤ੍ਰਿੰਞਣਾਂ 'ਚ ਕੱਤਦੀ ਦਾ ਰੂਪ ਕੋਈ ਖਾ ਗਿਆ
ਤੇਰੇ ਵਿਹੜੇ ਵਿਚ ਫਿਰਦੇ ਨੇ ਨਾਗ
ਓ ਸ਼ੇਰਾ ਜਾਗ
ਓ ਜੱਟਾ ਜਾਗ ।

ਖੋਹ ਕੇ ਨਾ ਲੈ ਜਾਣ ਫੇਰ ਕਿਤੇ ਹੇਣੀਆਂ
ਮਾਵਾਂ ਦੀਆਂ ਲੋਰੀਆਂ ਤੇ ਨੂੰਹਾਂ ਦੀਆਂ ਵੌਣੀਆਂ
ਭੈਣਾਂ ਦੀਆਂ ਚੁੰਨੀਆਂ ਤੇ ਵੀਰਾਂ ਦੀਆਂ ਘੋੜੀਆਂ
ਕਿਤੇ ਲੁਟੇ ਨਾ ਓ ਜਾਣ ਸੁਹਾਗ
ਓ ਸ਼ੇਰਾ ਜਾਗ
ਓ ਜੱਟਾ ਜਾਗ ।

ਸਹੁੰ ਤੈਨੂੰ ਲੱਗੇ ਤੇਰ ਮਹਿਲਿਆਂ ਦੇ ਬਾਗਾਂ ਦੀ
ਸਹੁੰ ਤੈਨੂੰ ਊਧਮਾਂ ਸਰਾਭਿਆਂ ਦੇ ਖ਼ਾਬ ਦੀ
ਰਖਣੀ ਏ ਸ਼ਾਨ ਬੀਬਾ ਤੂੰਹੀਓਂ ਹੀ ਪੰਜਾਬ ਦੀ
ਤੇਰੇ ਖਿੜਦੇ ਰਹਿਣ ਗੁਲਾਬ
ਓ ਸ਼ੇਰਾ ਜਾਗ
ਓ ਜੱਟਾ ਜਾਗ :

Figure 4.9: Document-6 in Gurmukhi Script

We have shown the document-6 in Gurmukhi script by figure 4.9. We have shown line and word segmentation of document-6 in Figure 4.10 after that Output of Line and Word segmentation of document-6 illustrate in Figure 4.11. We have also illustrated the result of line segmentation and word segmentation for different document by Table-4.2 and Table-4.3 respectively for both Devanagari script and Gurmukhi script. Here we have obtained the accuracy 100% at line segmentation level and 99.75% at word segmentation level. And for only Devanagari script we have illustrated the result of word, Characters and Top Characters Segmentation for five documents by Table-4.4. Hence we have obtained the accuracy 99.75% at word segmentation level, 98.89% at character segmentation level and 97.40% at top character segmentation level. This is better than previous results.

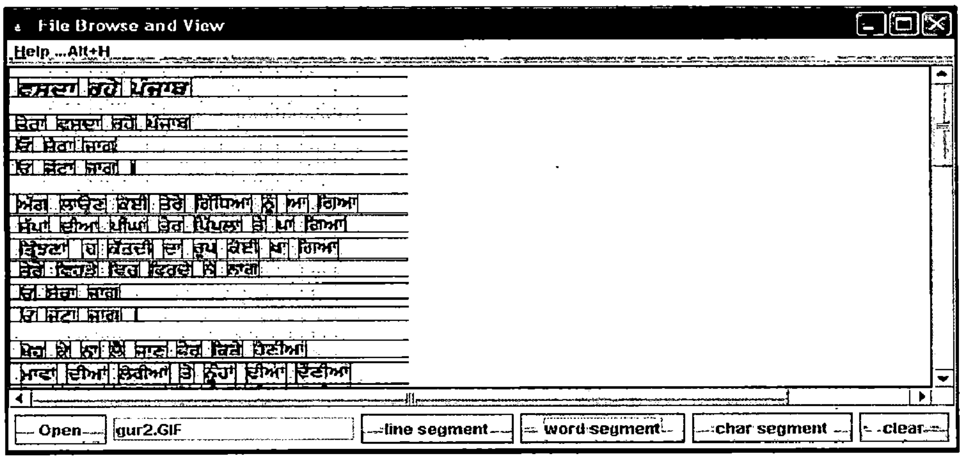
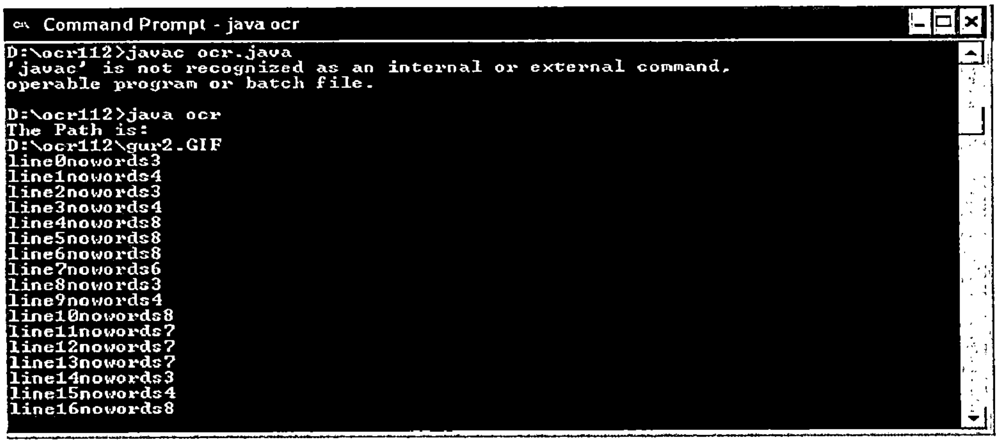


Figure 4.10: Line and Word segmentation of document 6



```
D:\ocr112>javac ocr.java
'javac' is not recognized as an internal or external command,
operable program or batch file.

D:\ocr112>java ocr
The Path is:
D:\ocr112\gur2.GIF
line0nowords3
line1nowords4
line2nowords3
line3nowords4
line4nowords8
line5nowords8
line6nowords8
line7nowords6
line8nowords3
line9nowords4
line10nowords8
line11nowords7
line12nowords7
line13nowords7
line14nowords3
line15nowords4
line16nowords8
```

Figure 4.11: Output of Line and Word segmentation for document 6

Table 4.2: Result of Line Segmentation by our Proposed Technique

Document	**No. of line**	**Correct Detected**	**Incorrect Segmentation**	**Accuracy**
Document 1	13	13	0	100%
Document 2	9	9	0	100%
Document 3	8	8	0	100%
Document 4	15	15	0	100%
Document 5	12	12	0	100%
Document 6	22	22	0	100%
Document 7	24	24	0	100%
Document 8	20	20	0	100%
Document 9	17	17	0	100%
Document 10	16	16	0	100%

Table 4.3: Result of Word Segmentation by our Proposed Technique

Document	**No. of words**	**Correct Detected**	**Incorrect segmentation**	**Accuracy**
Document 1	68	68	0	100%
Document 2	118	117	2	99%
Document 3	42	42	0	100%
Document 4	90	90	0	100%
Document 5	87	87	0	100%
Document 6	120	120	0	100%
Document 7	104	104	0	100%
Document 8	98	97	1	99%
Document 9	56	56	0	100%
Document 10	103	102	1	99%

Table 4.4: Result of word, Characters and Top Characters Segmentation for Devanagari Script Document.

Document	**No. of words**		**No. of Characters**		**No. of Top Characters**	
	No. of original words	**No. of recognize words**	**No. of original characters**	**No. of recognize words**	**No. of original Top characters**	**No. of recognize Top characters**
Document-1	68	68	182	180	59	58
Document-2	118	117	262	259	102	99
Document-3	42	42	102	101	35	34
Document-4	90	90	215	213	79	77
Document-5	87	87	228	225	72	70
Total	405	404	989	978	347	338
Accuracy	99.75%		98.89%		97.40%	

We have done comparison of line segmentation for printed Devanagari script and Gurmukhi script from Jindal and Sharma method [24] with our method which shown in Table 4.5 and in Figure 4.12. Hence we have obtained the accuracy 100% at line segmentation level for Devanagri script and Gurmukhi script for above documents.

Table 4.5: Comparison Accuracy in line segmentation

Method	Accuracy for line segmentation	
	For Devanagari script	For Gurmukhi script
Jindal and Sharma method	99.79%	98.12%
Our method	100%	100%

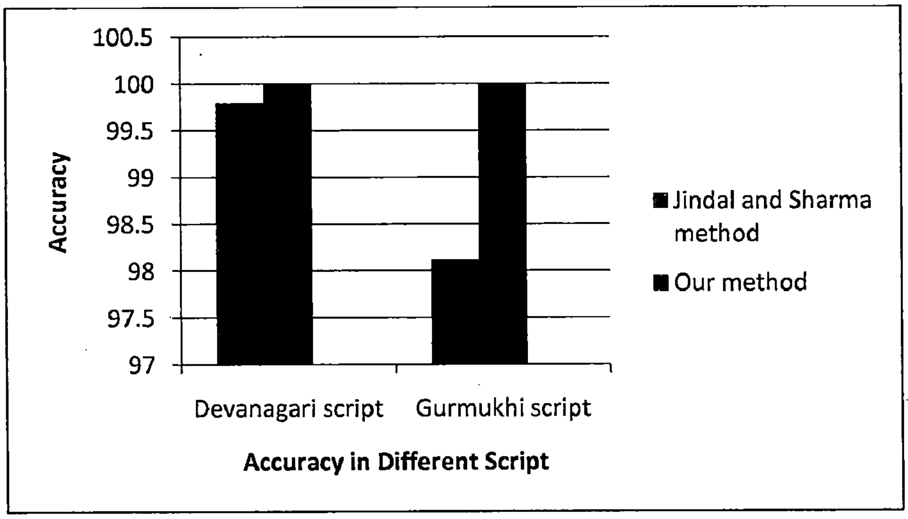
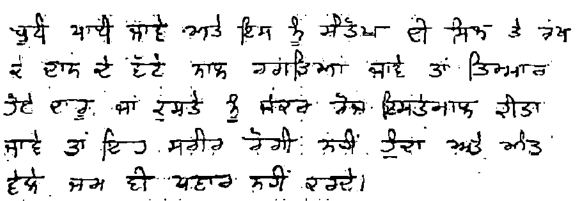


Figure 4.12: Accuracy in line segmentation

We have done comparison from Sharma and Singh (SS) method [31] for Gurmukhi script with our method . For comparison, we have considered four documents in handwritten Gurmukhi script.In Figure 4.14 and Figure 4.15, we have illustrated line and word segmentation of Document11 and Output of Line and word segmentation of Document 11 respectively.And overall results shown by SS method and our method for same document 11 in Figure 4.16 with the help of Table 4.6 and Table 4.7 .Comparison between SS method and our method shown in Table 4.8.



ਚੂਧੀ ਪਾਈ ਜਾਵੇ ਅਤੇ ਇਸ ਨੂੰ ਸੰਤੋਖ ਦੀ ਸਿਲ ਤੇ ਰਖ
ਦੇ ਦਾਨ ਦੇ ਦੰਦੇ ਨਾਲ ਰਗੜਿਆ ਜਾਵੇ ਤਾਂ ਤਿਆਰ
ਹੋਏ ਦਾਰੂ ਜਾਂ ਦਸਤੇ ਨੂੰ ਸੰਦਰ ਹੋਸ ਇਸਤੇਮਾਲ ਕੀਤਾ
ਜਾਵੇ ਤਾਂ ਇਹ ਸਰੀਰ ਰੋਗੀ ਨਹੀਂ ਹੁੰਦਾ ਅਤੇ ਮੌਤ
ਵੇਲੇ ਜਮ ਦੀ ਪਕੜ ਨਹੀਂ ਕਰਦੇ।

Figure 4.13: Document 11 in Gurmukhi script

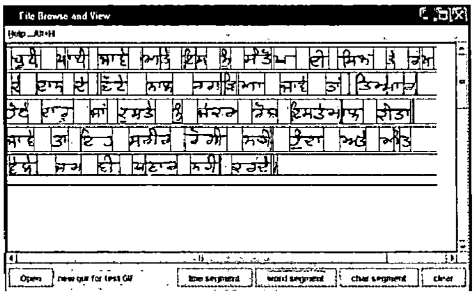
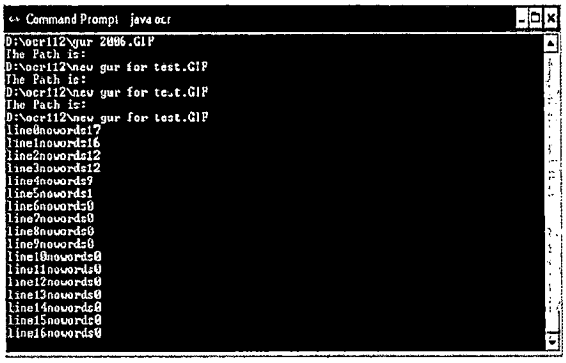


Figure 4.14: Line and word segmentation of Document 11



```
D:\ocr112\gur 2006.GIF
The Path is:
D:\ocr112\new gur for test.GIF
The Path is:
D:\ocr112\new gur for te.t.GIF
The Path is:
D:\ocr112\new gur for test.GIF
line0nowords17
line1nowords16
line2nowords12
line3nowords12
line4nowords9
line5nowords1
line6nowords0
line7nowords0
line8nowords0
line9nowords0
line10nowords0
line11nowords0
line12nowords0
line13nowords0
line14nowords0
line15nowords0
line16nowords0
```

Figure 4.15: Output of Line and word segmentation of Document 11

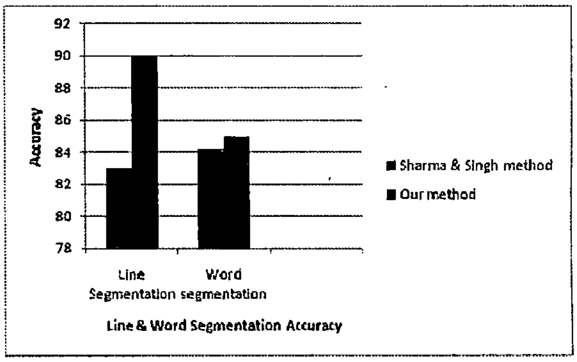


Figure 4.16: Accuracy for line and word Segmentation

Table 4.6: Results for Line segmentation

Document	No of Lines	Correctly Detected	Inaccurate segmentation	Accuracy
Document 11	5	6	1	83.34%
Document 12	10	9	1	90%
Document 13	20	19	1	95%
Document 14	12	11	1	91%

Table 4.7: Results for Word segmentation

Document	No. of words	Correctly Detected	Inaccurate segmentation	Accuracy
Document11	57	67	10	85%
Document12	85	74	11	87%
Document13	98	83	15	84%
Document14	74	63	11	84.5%

Table 4.8: .Comparison Accuracy between SS method and our method

Methods	Overall Accuracy	
	Line segmentation	Word segmentation
Sharma and Singh method	83.02%	84.17%
Our method	90%	85%

Hence we have obtained the accuracy 90% at the level of line segmentation and 85% at the level of word segmentation for handwritten gurmukhi script.

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In this Dissertation, we have done worked in two parts. In first part, we have proposed algorithm and implementation for text segmentation from image documents based on Daubechies wavelet and 2-mean classification .Our method is accurate from latest D-Tree method, Haar wavelet method [8, 9] and Naive Bayes Classifier method [7]. And we have also illustrated accuracy comparison from these methods [7, 8, and 9] in implementation section.

In second part, we have presented a modified algorithm for segmentation of line, word, character, top character for Devanagari Script and also Segmentation of line, word for Gurmukhi Script. A performance of 100% at line level, approximately 100% at word level, 99% at character level, and 97% at top character level for printed Devanagari script is obtained. And also performance of 100% at line level and 99% at word level for printed Gurmukhi script is obtained. And we have obtained the accuracy 90% at the level of line segmentation and 85% at the level of word segmentation for handwritten gurmukhi script. The overall successful segmentation achieved through the proposed algorithm is better than previous result.

5.2 Suggestion for Future Work

The proposed algorithm for segmentation text into line, word, and character gives adequate amount of scope for extension. Since at few point segmentation was good but at few point it was not up to the expectations. This may be because of the shape of characters. All these issues can be dealt in the future for printed documents in Devanagari and Gurumukhi script by making few changes to proposed work. The proposed method for text segmentation from image document gives adequate amount of scope for extension. In the future, additional efforts on both the theoretical and the practical side need to be made on at least the following points:

– Improved separation of graphic linked to text

– Accuracy and segmentation rate will be improved.

REFERENCES

[1] D. Chen, H. Bourlard and J. Thiran, "Text Identification in Complex Backgrounds Using SVM", Proc. of the International Conf. On Computer Vision and Pattern Recognition, Chen Bourlard, H. Thiran ,pp. 621-626, 8-14 Dec. 2001.

[2] M. Pietikäinen and O. Okun, "Text Extraction from Grey Scale Page Images by Simple Edge Detectors", Proc. of the 12th Scandinavian Conf. On Image Analysis, Bergen, Norway, pp. 628-635, 11-14 June 2001.

[3] Jie Xi, Xian-Sheng Hua, Xiang-Rong Chen, et al., "A Video Text Detection and Recognition System", Proc. of ICME 2001, Waseda University, Japan, pp. 1080-1083, August 2001.

[4] Q. Yuan and C. L. Tan, "Page Segmentation and Text Extraction from Grey-Scale Images in Micro Film Format", SPIE Proc. on Document Recognition and Retrieval, vol. 4307, pp.323-332, 2000.

[5] H. Choi and R. G. Baraniuk, "Multiscale Image Segmentation Using Wavelet-Domain Hidden Markov Models", IEEE Transactions on Image Processing, vol. 10(9), pp. 1309-1321, Sep. 2001.

[6] Shulan Deng and Shahram Latifi, "Fast Text Segmentation Using Wavelet for Document Processing", Proc. of the 4th WAC, ISSCI, IFMIP, Maui, Hawaii, USA, pp. 739-744, 11-15 June 2000.

[7] M. M. Haji, S. D. Katebi, "An Efficient Text Segmentation Technique Based on Naive Bayes Classifier", GVIP Journal, Volume 5, Issue 7, July 2005

[8] M. M. Haji, S. D. Katebi, "Machine Learning Approaches to Text Segmentation", Scientia Iranica, Vol. 13, No. 4, pp 395-403, October 2006.

[9] Machine Learning Project, "Text Segmentation Using Decision Trees".[Online], Available: http://pasargad.cse.shirazu.ac.ir/~mhaji/ml2/Project2.html. [Last accessed 10 March 2010].

[10] A. Zahour, B. Taconet, P. Mercy, and S. Ramdane, "Arabic Hand-written Text-line Extraction", Proceedings of the Sixth International.Conference on Document Analysis and Recognition, ICDAR, Seattle, USA, pp. 281–285, 13 September 2001.

[11] N. Tripathy and U. Pal, "Handwriting Segmentation of unconstrained Oriya Text", in the proceedings of International Workshop on Frontiers in Handwriting Recognition, pp. 306–311, 26-29 Oct 2004.

[12] G. Louloudis, B. Gatos, I. Pratikakis and K. Halatsis, "A Block Based Hough Transform Mapping for Text Line Detection in Handwritten Documents", in the proceedings of Tenth International Workshop on Frontiers in Handwriting Recognition, La Baule, pp. 515-520, October 2006.

[13] Y. Li, Y. Zheng, D. Doermann, and S. Jaeger, "A new algorithm for detecting text line in handwritten document", in the proceedings of International Workshop on Frontiers in Handwriting Recognition, pp. 35–40, October 2006.

[14] L. Likforman-Sulem and C. Faure, "Extracting text lines in handwritten documents by perceptual grouping", Advances in handwriting and drawing: a multidisciplinary approach, C. Faure, P. Keuss, G. Lorette and A. Winter Eds, Europia, Paris, pp. 117-135, 1994.

[15] I.S.I. Abuhaiba, S. Datta and M. J. J. Holt, "Line Extraction and Stroke Ordering of Text Pages", in the Proceedings of Third International Conference on Document Analysis and Recognition, Montreal, Canada, pp. 390-393, 14-17 August 1995.

[16] C. Weliwitage, A. L. Harvey and A. B. Jennings, "Handwritten Document Offline Text Line Segmentation", in the Proceedings of Digital Imaging Computing: Techniques and Applications, pp. 184-187, 2005.

[17] A. Zahour, B. Taconet, L. Likforman-Sulem and Wafa Boussellaa, "Overlapping and multi-touching text-line segmentation by Block Covering analysis", Pattern analysis and applications, Vol . 12, No. 4, pp. 335-351, 9 July 2008.

[18] X. Wang, V. Govindaraju, S. N. Srihari, "Holistic Recognition of Handwritten Character Pairs", Pattern Recognition, vol. 33, pp. 1967-1973, 2000.

[19] Y. Ariki, Y. Mot, "Segmentation and Recognition of Handwritten Characters using Subspace Method", Proc. 3rd ICDAR, Vol. 1, pp. 120-123, 14-16 Aug. 1995.

[20] Satadal Saha, Subhadip Basu, Mita Nasipuri and Dipak Kr. Basu, "A Hough Transform based Technique for Text Segmentation" , Journal of Computing, Vol. 2, Issue 2, February 2010.

[21] V. Bansal, R.M.K. Sinha, "Segmentation of Touching and Fused Devanagari Characters", Pattern Recognition, vol. 35, pp. 875-893, April 2002.

[22] U. Garain, B. B. Chaudhuri, "Segmentation of Touching Characters in Printed Devnagari and Bangla Scripts using Fuzzy Multifactorial Analysis", Proc. 6th ICDAR, pp. 805-809, 10-13 Sept. 2001.

[23] Bidyut B. Chaudhuri, Sumedha Bera, "Handwritten Text Line Identification In Indian Scripts", 10th International Conference on Document Analysis and Recognition, 26-29 July 2009.

[24] M.K. Jindal, R.K. Sharma, G.S. Lehal, "Segmentation of Horizontally Overlapping Lines in Printed Indian Scripts" , International Journal of Computational Intelligence Research.ISSN 0973-1873 Vol.3, No.4 , pp. 277–286, June 2007

[25] Naresh Kumar Garg, Lakhwinder Kaur, M. K. Jindal, "Segmentation of Handwritten Hindi Text", International Journal of Computer Applications (0975 – 8887) Volume 1, No. 4, March 2010.

[26] Internet: http://www.eng.auburn.edu/~sjreeves/Classes/IP/IP.html [Last accessed: June 17, 2010].

[27] Internet: http://encyclopedia2.thefreedictionary.com/image+processing [Last accessed: June 17, 2010].

[28] Internet: http://en.wikipedia.org/wiki/Daubechies_wavelet [Last accessed: June 17, 2010].

[29] Internet: http://databases.about.com/od/datamining/a/kmeans.htm [Last accessed: June 17, 2010].

[30] Internet:http://www.mathworks.com/access/helpdesk_r13/help/toolbox/images/morph4.html [Last accessed: June 17, 2010].

[31] Rajiv K. Sharma & Dr. Amardeep Singh, "Segmentation of Handwritten Text in Gurmukhi Script", International Journal of Computer Science and Security, volume (2), issue (3), 2006.