Skew Detection, Correction and Segmentation of Handwritten Kannada Document

Mamatha Hosalli Ramappa[1] and Srikantamurthy Krishnamurthy[2]

[1]*Department of Information Science and Engineering*
P E S Institute of Technology, Bangalore, India

[2]*Department of Computer Science and Engineering*
P E S Institute of Technology (South Campus), Bangalore, India

Abstract

Optical character recognition (OCR) refers to a process of generating a character input by optical means, like scanning, for recognition in subsequent stages by which a printed or handwritten text can be converted to a form which a computer can understand and manipulate. A generic character recognition system has different stages like noise removal, skew detection and correction, segmentation, feature extraction and classification. Results of the later stages can affect the performance of the subsequent stages in the OCR process. To make the results of the subsequent stages more accurate, the skew detection and correction and segmentation play an important role. In this paper, we have proposed schemes for skew detection and correction, segmentation of handwritten Kannada document using bounding box technique, Hough transform and contour detection respectively. An average segmentation rate of 91% and 70% for lines and words is obtained respectively.

Keywords: *OCR, Skew detection and correction, segmentation and Handwritten Kannada document*

1. Introduction

The automation process involves the document image analysis (DIA) that concerns with the automatic interpretation of document into text, graphics, drawings etc and an Optical Character Recognition (OCR) system, which involves the process of transforming human readable and optically sensed data to machine understandable codes. The high performance of any recognition system (OCR systems) depends on the detailed analysis of preprocessing and segmentation operations for removing noises and extracting character components respectively from the input document image [1].

Skew angle estimation and correction of a document page is an important task for document analysis and OCR applications. During digitization of documents, it often happens that the document page is not aligned correctly. The relative inclination angle of the page being acquired must be detected and accounted for as it can cause serious performance deterioration of segmentation and recognition stage of any text processing system.

Segmentation is the process of extracting objects of interest from an image. The first step in segmentation is detecting lines. The subsequent steps are detecting the words in each line and the individual characters in each word. This 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. Segmentation of words into individual letters has been one of the major problems in handwriting recognition. Despite several successful works all over the world, development of such tools in specific languages is still an ongoing process especially in the Indian context. The complexity involved in the segmentation of characters in the uneven spacing between text lines and adjacent characters. The text lines can also be skewed in some cases.

From the literature survey it is clear that most of the work has been done for English, Chinese and Arabic etc. Few works are reported on Indian languages like Bangla, Devanagari, Assamese and Telugu scripts. Very few works are reported on text line extraction on Handwritten Kannada Script and on skew detection of Kannada document. To the best of our knowledge there has been no work in the word and character segmentation of the handwritten Kannada script and in the skew correction. Skew angle estimation and correction of a Kannada document page is an important task for document analysis and OCR applications .Segmentation of handwritten Kannada script into lines, words and character is of great importance and much demanded by some specific applications. Segmentation of handwritten Kannada script poses challenges due to additional modifier characters, writing styles, skewed lines, inter and intra word gaps. This motivated us to design effective schemes for detection and correction of the skew angle, segmentation of the handwritten Kannada document which can be used in the later stages of OCR so that the performance of the subsequent steps in document image analysis would be more accurate.

In this paper we have proposed a methodology based on bounding box for skew detection and correction, Hough transform for segmentation of the handwritten Kannada script into lines and contour detection for word segmentation. The rest of the paper is organized as follows. Section 2 describes the characteristics of Kannada script, Section 3 and 4 discusses about the skew detection, correction and segmentation respectively. Section 5 briefly discusses the experimental setup and the results obtained are discussed respectively. Finally in Sections 6 and 7, comparative study and conclusions are made.

2. The Characteristics of Kannada Script

In this section, we will briefly describe some of the main characteristics of Kannada script to point out the main difficulties for segmenting. Kannada is a popular script and it is the official language of the southern Indian state, Karnataka. Kannada is a Dravidian language mainly used by the people of Karnataka, Andhra Pradesh, Tamil Nadu and Maharashtra. Kannada is spoken by about 44 million people. The language has 47 characters in its alphabet set (13 vowels and 34 consonants).

ಅ ಆ ಇ ಈ ಉ ಊ ಋ ಎ ಏ ಐ ಒ ಓ ಔ

Figure 1. Vowels of Kannada Script

ಕ ಖ ಗ ಘ ಙ

ಚ ಛ ಜ ಝ ಞ

ಟ ಠ ಡ ಢ ಣ

ತ ಥ ದ ಧ ನ

ಪ ಫ ಬ ಭ ಮ

ಯ ರ ಲ ವ ಶ ಷ ಸ ಹ ಳ

Figure 2. Consonants of Kannada Script

A Character can be one of the following,

i. A stand alone vowel or a consonant

ii. A consonant modified by a vowel.

iii. A consonant modified by one or more consonants and a vowel.

Some of the complex characters are listed below to show the complication of the segmentation. The following figure shows the conjunct consonant (Subscript/Vatthu)

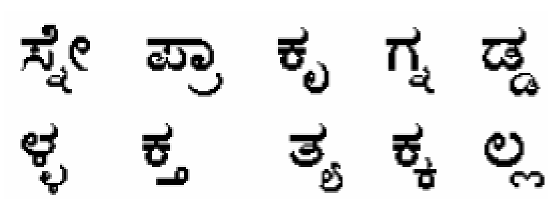


Figure 3. Shows the Conjunct Consonant (Subscript/Vatthu)

3. Skew Angle Detection and Correction

3.1 Related Work

Document skew has been recognized as a universal problem of document imaging. Hand placement or automatic document feeder mechanisms normally create 1-3^o of skew, due to the documents incorrect placement, or to a slight variation in roller speed. In some cases, the skew can reach as much as 10^o .When the skew angle is 2-3^o, the accuracy of document analysis and OCR is reduced. When it is more than 5^o however, the result becomes unreliable. As a result, the skew estimation and correction of document images needs to be carried out before segmentation and classification [2].

Many skew detection and correction techniques have been developed and they are mainly classified into Projection profile, Fourier method, Nearest neighbor clustering, Correlation, Hough transform technique respectively. Projection profile methods are the commonly used techniques, and usually work well for text-only documents. In this method, the projection profile is computed at a number of angles and a measure of difference of peak and trough height is made for each angle. The maximum difference corresponds to the best alignment with the text line direction, which, in turn, determines the skew angle. A modified projection profile method has been proposed in [3].

In the second method, the direction for which the density of the Fourier space is the largest gives the skew angle. And very often for a document image, the largest density direction of the Fourier space is on a vertical line and the true density direction may not be the largest. This makes the skew detection/search difficult. A method based on Fourier transform (FT) is presented in [4].Nearest neighbor clustering to skew detection is proposed in [5]. In this method, all the connected components in the document were found and computed the direction of its nearest neighbor for each component. A histogram of the direction angle is computed, the peak of which indicates the document skew angle.

In [6] a method for determining the skew angle of an image using cross-correlation between lines at a fixed distance is been introduced. It is based on the observation that the correlation between two vertical lines in an image of a skewed document is maximized in general if one line is shifted relatively to the other line such that the character base line levels for the two lines are coincident. Hough transform has been used in [7] for skew detection. The basic method consists of mapping points in Cartesian space (x, y) to sinusoidal curves in (ρ, θ) space via the transformation ρ: =x cos θ + y sin θ . Each time a sinusoidal curve intersects another at a particular value of ρ and θ, the likelihood increases that a line corresponding to that (ρ , θ) coordinate value is present in the original image. An accumulator array is used to count the number of intersection at various ρ and θ values. The skew is then determined by the θ values corresponding to the highest number of counts in the accumulator array.

The boundary growing approach to extract the lower and the uppermost coordinates of pixels of characters of text lines present in the document is proposed in [8]. In order to estimate skew angle for a document the method substitute the lower and upper most coordinates of pixels of characters in linear regression analysis (LRA). The proposed method is also used to determine skew angle for scaled documents. However, the method assumes that the space between the text lines is greater than the space between the words and characters. A method is specifically designed in [9] for Indian scripts like Devanagari and Bangla that have a head line (Shirorekha or Matra). The results of this skew detection methodology are proved to be comparable to those of Hough transform based methods but with less computation. Nevertheless, the method is completely dependent on the headlines, which join the characters in the word and makes the word appear as a single component.

A method is based on connected component blotching and linear regression is proposed in [10].An approach based on wavelets for document skew estimation is presented in [11]. Skew estimation of binary document images using static and dynamic thresholds are proposed in [12]. Estimation of skew angle in binary document images using connected component analysis and Hough transform is proposed in [13].

Skew detection of scripts having a head line (Shirorekha or Matra) is easier as the headlines are themselves sufficient to detect the skew angle. Detection of skew angle in English documents is much tougher when compared to Devanagari and Bangla scripts as English scripts do not have headline. Hence there are many methods in literature for skew detection in English scripts. Nevertheless, skew detection in English scripts, although they do not have any head line (Shirolekha or Matra) is relatively simpler compared to some of the Indian scripts like Kannada, Telugu etc., as English characters are free from vowels or consonant modifiers below a character and in addition, have a predominant base line. Therefore the methods developed for English text are not applicable to all Indian scripts especially for the scripts which are rich in vowels/consonant modifiers viz., Kannada, Telugu scripts [10]. This motivated us to design an effective scheme to detect the skew angle of the Kannada document, which can be used for skew correcting so that the performance of the subsequent steps in document image analysis would be more accurate.

Once the skew angle of a document image has been estimated, the image should then be corrected to generate a non-skew version. Skew correction methods are classified into pixel-oriented and contour-oriented. In pixel-oriented method, each pixel in the original image is rotated in the output image. These are categorized into direct and indirect methods depending on the type of rotational equation used. Both direct method and contour-oriented method have the rounding problem while indirect method does not have rounding problem but it is time consuming [2].

3.2 Proposed Methodology

The Skew Angle Detection and Correction is achieved through Bounding Box Technique. The technique used is very simple compared to the above said methods since the method do not involve any expensive methods like Hough transform and others to determine skew angles for documents.

Bounding Box technique is a way of finding the extreme corners of text image. If the four extreme points supposed to be found correctly forming a perfect rectangle, we can easily find out the estimated angle in much less time. The advantage of this Bounding box algorithm is that if any two of the four corner points detected correctly, it will give the accurate skewed angle. The bounding box of a geometric shape in 2D is the rectangle with the smallest area in a given orientation (usually upright) that complete contains the shape. The best fit bounding box is the smallest bounding box among all the possible orientations for the same shape. One of the applications of the best fit bounding boxes is the skew estimation from the text blocks in document images. This approach is capable of multi skew estimation and location, as well as being able to process documents with sparse text regions [14].

Skew correction is performed by rotating the document through an angle $-\theta$ with respect to the horizontal line, where the detected angle of skew is θ. In order to prevent the image being rotated off the image plane, the skewed image is first translated to the center and the new image dimensions are computed. The image for which the skew angle has to be found is preprocessed first. The original image is binarised. To know the skew of the document we are using the minimum bounding box technique. To compute the bounding box on an image, first we extract all the positions of the white pixels and store it in a vector of points. These vectors of points are going to be converted in the form of matrix. Next we compute the minimum area bounding rectangle (possibly rotated) for the specified point set and find the center, size and the angle by which it is rotated.

After the skew angle estimation, the skew correction of the document has to be done. Skew correction approaches can be categorized into direct, indirect and contour-oriented [2].Here we have used the direct method where given the skew angle θ, the direct method de-skews the image by rotating the black pixels by $(-\theta)$[15].A black pixel p in the input image is transformed into p' by multiplying the coordinates of p by a rotational matrix as in (1):

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} \quad (1)$$

Where $(x, y)'$ are the coordinates of p in the input image and $(x', y')'$ are the coordinates of p' in the output image.

After we have the rotated bounding box we apply the geometric transformations i.e., affine transformation to deskew it. An affine transformation is an important class of linear 2D geometric transformations which maps variables (e.g., pixel intensity values located at position(x1, y1) in an input image) into new variables (e.g., (x2,y2) in an output image) by applying a linear combination of translation, rotation, scaling and/or shearing(i.e., non-uniform scaling in some directions) operations. Next we crop the image in order to remove borders. If the skew angle is positive, the angle of the bounding box is below -45 degrees because the angle is given by taking as a reference a "vertical rectangle" i.e., with the height greater than the width. If the angle is positive we swap the height and width before cropping the image. The bounding box technique won't work well with large angle. In order to test the proposed methodology we had collected documents with varying skew angles. The proposed work is able to detect the skew angles between the ranges +45 degrees to -45 degree.

4. Segmentation

4.1 Related Work

In the recent past, the number of document images available for Indian languages has grown drastically with the establishment of Digital Library of India. The digital library documents originate from a variety of sources, and vary considerably in their structure, script, font, size, quality, etc. Text line extraction from unconstrained handwritten documents is a challenge because the text lines are often skewed and the space between lines is not obvious. The complexity involved in the segmentation of the handwritten documents for Indian languages like Telugu, Tamil and Malayalam is very well explained in [16]. Curved and non-parallel text lines in handwritten documents also make the segmentation and recognition challenging.

Handwriting text line segmentation approaches can be categorized according to the different strategies used. These strategies are projection based, smearing, grouping, Hough-based, graph-based and Cut Text Minimization (CTM) approach[17].The projection-based algorithm proposed by Arivazhagan, et. al., [18] first obtains an initial set of candidate lines from the piece-wise projection profile of the document .The lines traverse around any obstructing handwritten connected component by associating it to the line above or below. The proposed method is robust to handle skewed documents and touching lines. In smearing based approach technique, consecutive black pixels along the horizontal direction are smeared. If the distance between the white space is within a predefined threshold, it is filled with black pixels. The bounding boxes of the connected components in the smeared image are considered as text lines. Li, et. al., [19] proposed a new approach for text line detection by adopting a state-of-the-art image segmentation technique. They first convert a binary image to gray scale using a Gaussian window, which enhances text line structures. Text lines are extracted by evolving an initial estimate using the level set method.

Grouping approach involves building alignments by aggregating units in a bottom-up approach. Units such as pixels, connected components, or blocks are then joined together to form alignments. Likforman-Sulem and Faure [20] proposed an approach based on perceptual grouping of connected components of black pixels. Text lines are iteratively constructed by grouping neighboring connected components based on certain perceptual criteria such as similarity, continuity and proximity. According to the authors the proposed technique cannot be used on degraded or poorly structured documents, such as modern authorial manuscripts.

In Hough-based approach, the Hough transform is used for locating straight lines in images. In [21] an iterative hypothesis validation strategy based on Hough transform was proposed. The skew orientation of handwritten text lines is acquired by applying the Hough transform to the center of gravity of each connected component in the document image. This technique is able to detect text line in handwritten documents which may contain lines oriented in different directions, erasures and annotations between main lines.

Sesh Kumar, et. al., [16] presented a graph cut based framework using a swap algorithm to segment document images containing complex scripts such as in Indian languages. In [22], the CTM method finds a path or cut line in between the text lines to be separated which minimizes the text line pixels cut by the segmentation line, especially descenders from the upper line and ascenders from the lower line. Sarkar, et. al., [23], proposed the bottom up approach of line segmentation from handwritten text. Roy et al [24], proposed morphology based handwritten line segmentation using foreground and back ground information. The morphological operation and run-length smearing algorithm (RLSA) is used. A method for line segmentation of handwritten Hindi text is reported in [25]. The method is based on header line detection, base line detection and contour technique. Basu, et, al., [26], presents text line extraction from multi-skewed handwritten documents. They assume that hypothetical water flows, form left and height sides of the image frame, face obstruction from characters of text lines. The stripes of areas left unwetted on the image frame are finally labeled for extraction of text lines.

For word segmentation there exist two distinct tendencies. In the first, after taking as input a text line image, the connected components are calculated. The distances between adjacent connected components are measured using a metric such as the Euclidean distance, the bounding box distance or the convex hull metric. Finally, a threshold is defined which is used to classify the calculated distances as either inter-word or inter-characters gaps [27]. In [28], the word segmentation problem is considered as a text line recognition task, adapted to the characteristics of segmentation. That is, at a certain position of a text line, it has to be decided whether the considered position belongs to a letter of a word, or to a space between two words. For this purpose, three different recognizers based on Hidden Markov Models are designed, and results of writer- dependent as well as writer-independent experiments are reported in the paper.

An approach based on fringe maps to generate segmenting paths between adjacent text lines is proposed in [29]. First they generate a fringe map for the input binary image; next the authors compute peak fringe numbers (PFN) to locate potential regions to find a separating path. PFNs between lines are used to generate a segmenting path to separate adjacent lines. In [30] method for line segmentation of handwritten Hindi text is presented. The method is based on header line detection, base line detection and contour following technique. No preprocessing like skew correction, thinning or noise removal has been done on the data. The authors claim that this method is suitable for fluctuating lines or variable skew lines of text. Also, they confirm that this method is invariant of non uniform skew between words in a line (non uniform text line skew) and the contour following after header line detection correctly separates some of the overlapped lines of text.

An approach to segment the scanned document image is presented in [31]. Here the whole image is considered as one large window. Then this large window is broken into less large windows giving lines, once the lines are identified then each window consisting of a line is used to find a word present in that line and finally to characters. The authors have used the concept of variable sized window, that is, the window whose size can be adjusted according to needs. In [32], the authors have proposed a novel two stage evaluation methodology for word segmentation techniques. They have proposed a robust evaluation methodology that treats the distance computation and the gap classification stages independently.

From the above literature survey it is clear that most of the work has been done for English, Chinese and Arabic etc. Few works are reported on Indian languages like Bangla, Devanagari, Assamese and Telugu scripts. Very few works are reported on text line extraction on handwritten Kannada script. To our best knowledge there has been no work in the word segmentation of the handwritten Kannada script. Segmentation of handwritten Kannada script into lines, words and character is of great importance and much demanded by some specific applications. Segmentation of handwritten Kannada script poses challenges due to additional modifier characters, writing styles, skewed lines, inter and intra word gaps.

4.2 Proposed Methodology

In this section, we propose a segmentation of unconstrained handwritten Kannada script into lines and words. The line segmentation method consists of two stages. the first stage is a preprocessing stage where we first binarise the original image and then we apply Sobel edge detector for edge detection and in the next stage we apply the Hough transform algorithm for line detection. Segmentation of the script into words also consists of two stages .In the first stage the image is preprocessed by applying binarisation, erosion and dilation and contour detection technique is applied in the next stage.

4.2.1 Line Segmentation

4.2.1.1 Sobel Edge Detector: The Sobel operator is used in image processing, particularly within edge detection algorithms. Technically, it is a discrete differentiation operator, computing an approximation of the gradient of the image intensity function. At each point in the image, the result of the Sobel operator is either the corresponding gradient vector or the norm of this vector. The Sobel operator is based on convolving the image with a small, separable, and integer valued filter in horizontal and vertical direction and is therefore relatively inexpensive in terms of computations.

The operator calculates the gradient of the image intensity at each point, giving the direction of the largest possible increase from light to dark and the rate of change in that direction. The result therefore shows how "abruptly" or "smoothly" the image changes at that point and therefore how likely it is that part of the image represents an edge, as well as how that edge is likely to be oriented. In practice, the magnitude (likelihood of an edge) calculation is more reliable and easier to interpret than the direction calculation.

Mathematically, the operator uses two 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical [33].

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

4.2.1.2 Hough Transform: In automated analysis of digital images, a frequently arising problem is detecting the simple shapes like straight line, circle or ellipse. In most of the cases an edge detector can be used as a pre-processing stage to obtain image points or image pixels that are on the desired curve in the image space. But due to imperfections in either the image data or the edge detector there may be missing or isolated or disjoint points or pixels on the desired curves as well as there may be spatial deviations between the ideal lines or circle or ellipse and the noisy edge points as obtained from the edge detector. For these reasons, it is often non-trivial to group the extracted edge features to an appropriate set of lines, circles or ellipses. The purpose of the Hough transform is to address this type of problem by making it possible to perform groupings of edge points into object candidates by performing an explicit voting procedure over a set of parameterized image objects[34] .That is the basic idea is 'each straight line in an image can be described by an equation and each white point if considered in isolation could lie on an infinite number of straight lines. In the Hough transform each point votes for every line it could be on .The lines with the most votes win' [35].

The HT was introduced by Paul Hough in a patent filed in 1962. It was used to detect curves in bubble chamber photographs and was brought to the attention of the mainstream image processing community by Rosenfeld. The Hough transform, HT, was first introduced as a method of detecting complex patterns of points in binary image data. It achieves this by determining specific values of parameters which characterize these patterns [36].

The Hough transform is briefly described below. Let us consider a single isolated edge point (x, y) in the image plane. There could be an infinite number of lines that could pass through this point. Each of these lines can be characterized as the solution to some particular equation. In the simplest form a line can be expressed in the slope-intercept form as y = mx + c where, m is the slope of the line with respect to x axis and c is the intercept on y axis made by the line. Any line can be characterized by these two parameters pair (m, c). For all the lines that pass through a given point (x, y), there is a unique value of c for m, given by (2)

$$c = y - m.\,x \tag{2}$$

Using slope-intercept parameters could make application complicated since both parameters are unbounded. As lines get more and more vertical, the magnitudes of m and c grow towards infinity. For computational purposes, however, it is better to parameterize the lines in the Hough transform with two other parameters, commonly called ρ(rho) and θ (theta), The parameter ρ represents the distance between the line and the origin, while θ is the angle of the vector from the origin to this closest point. Using this parameterization, the equation of the line can be written as $\rho{:} = x\cos\theta + y\sin\theta$. It is therefore possible to associate to each line of the image, a couple (ρ, θ) which is unique if $\theta \epsilon [0,2\pi]$ and $\rho > 0$. The (ρ, θ) plane is sometimes referred to as Hough space for the set of straight lines in two dimensions.

The set of (m, c) values corresponding to the line passing through point (x, y) form a line in (m, c) space. Every point in image space (x, y) corresponds to a line in parameter space (m, c) and vice versa. The Hough transform works by letting each feature point (x, y) vote in (m, c) space for each possible line passing through it. These votes are totaled in an accumulator. Accumulator is an array used by the Hough transform algorithm to detect the existence of a line $y = mx + c$. The dimension of the accumulator is equal to the number of unknown parameters of the Hough transform problem. For example, the linear Hough transform problem has two unknown parameters: the pair (m,c) or the pair (r,θ). The two dimensions of the accumulator array would correspond to quantized values for (r,θ). For each pixel and its neighborhood, the Hough transform algorithm determines if there is enough evidence of an edge at that pixel. If so, it will calculate the parameters of that line, and then look for the accumulator's bin that the parameters fall into, and increase the value of that bin. By finding the bins with the highest values, typically by looking for local maxima in the accumulator space, the most likely lines can be extracted, and their (approximate) geometric definitions read off. The simplest way of finding these peaks is by applying some form of threshold, but different techniques may yield better results in different circumstances - determining which lines are found as well as how many. Since the lines returned do not contain any length information, it is often next necessary to find which parts of the image match up with which lines. Moreover, due to imperfection errors in the edge detection step, there will usually be errors in the accumulator space, which may make it non-trivial to find the appropriate peaks, and thus the appropriate lines.

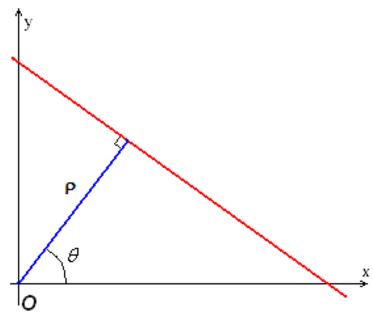


Figure 4. Representation of Straight Line in (ρ, θ) Format

4.2.2 Word Segmentation

The spacing between the words is used for word segmentation. For Kannada script, spacing between the words is greater than the spacing between characters in a word. Contour Detection is employed for word detection.

A contour is a list of points that represent, in one way or another, a curve in an image. This representation can be different depending on the circumstance at hand. There are many ways to represent a curve. Contours are represented by sequences in which every entry in the sequence encodes information about the location of rectangle of the contour. And the rectangle is drawn on each contour to obtain words segmented image.

The words segmentation approaches in printed/handwritten text lines are usually based on various heuristics and assumption that the gaps between words (inter-word gap) are larger than those inside the words (intra-word gap). Words extraction from handwritten text lines usually involves calculation of threshold for inter-word gap. Hence, the text line is decomposed into series of single connected or horizontally overlapped connected components. Following decomposition, a threshold distance is determined to decide the gap either inter-word or intra-word, finally the words are extracted.

The first goal we must achieve is extracting the contours belonging to the image to be processed, which will give information about the contours existing in our image. With the set of contours detected, we can draw the contour of the object or detect the bounding area.

Algorithm for word segmentation:

1. Initialize the contour variables, contour and contourLow to zero.
2. Create storage for Contour Detection.
3. Search for the contours in the image.
4. For each contour found, detect the bounding rectangle of the contour.
5. The rectangle is drawn on the contour.

5. Experimental Results

This section presents the results of the experiments conducted to study the performance of the proposed method. The method has been implemented in the OpenCV 2.1 on Dual Core 3GHZ with 2GB RAM. For the experiment, we have considered 40 document images collected from different people of various age group and professions. The handwritten Kannada document is then scanned by flatbed scanner at a resolution of 300 dpi. The data set contains varieties of writing styles. Non-text elements are not included in the documents and almost all the documents have two or more adjacent text lines touching in several areas. Some of the documents have variable skew angles among text lines with different skew directions. We have considered single column document pages for the experimentation. The number of lines in each document varies from 10 to 21 lines. For the skew detection experimentation, the documents were tilted by a prespecified angle ranging between 0^o and $\pm 45^o$. The proposed work is able to detect and correct the skew angles between the ranges +45 degrees to -45 degree. Figure 5 shows the results of the proposed methodology for the documents with different skew angles.

In the segmentation experimentation, the Line Segmentation accuracy of 40 text documents is measured by the fraction percentage of number of lines correctly segmented to the total number of lines present in the document. The average accuracy obtained is 91%. Word Segmentation accuracy of 40 text documents is measured by the fraction percentage of number of words correctly segmented to the total number of words present in the document. The average accuracy obtained is 70%.Most of the errors encountered in the word segmentation phase are due to the non-uniform spacing between characters of the same word and between adjacent words. Different stages from input handwritten Kannada document image to the segmentation at respective levels are shown from Figure 6.

6. Comparative Study

The Table 1 and 2 show the comparison of existing methods with proposed method for skew detection and line segmentation respectively. To compare our proposed method with the existing work is very difficult as very few works exist in the skew detection and line segmentation of handwritten Kannada script which is experimented on different datasets of various complexities. To the best of our knowledge there is no work found in the skew correction and the word segmentation of the handwritten

Kannada script document. In [37] the authors have experimented with the methods specified in [38] and [39].

7. Conclusion

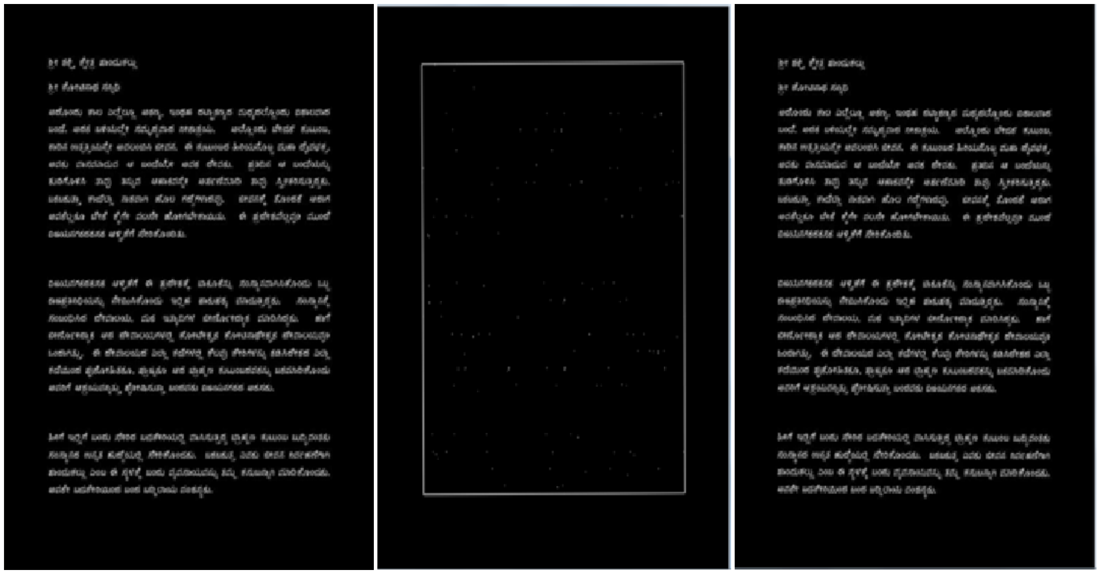
In this paper, we have proposed a methodology for skew detection, correction and segmentation of handwritten Kannada document into lines and words. Bounding box technique is used for the skew detection and correction. For segmentation of document into lines and words Hough transform and contour detection is used respectively. The method was tested on totally unconstrained handwritten Kannada scripts, which pays more challenge and difficulty due to the complexity involved in the script.91% of line segmentation rate is obtained and because of the varying inter and intra word gaps we could get average segmentation rate of 70% for words.

Table 1. Comparison of Proposed Method with the Existing Methods for Skew Detection

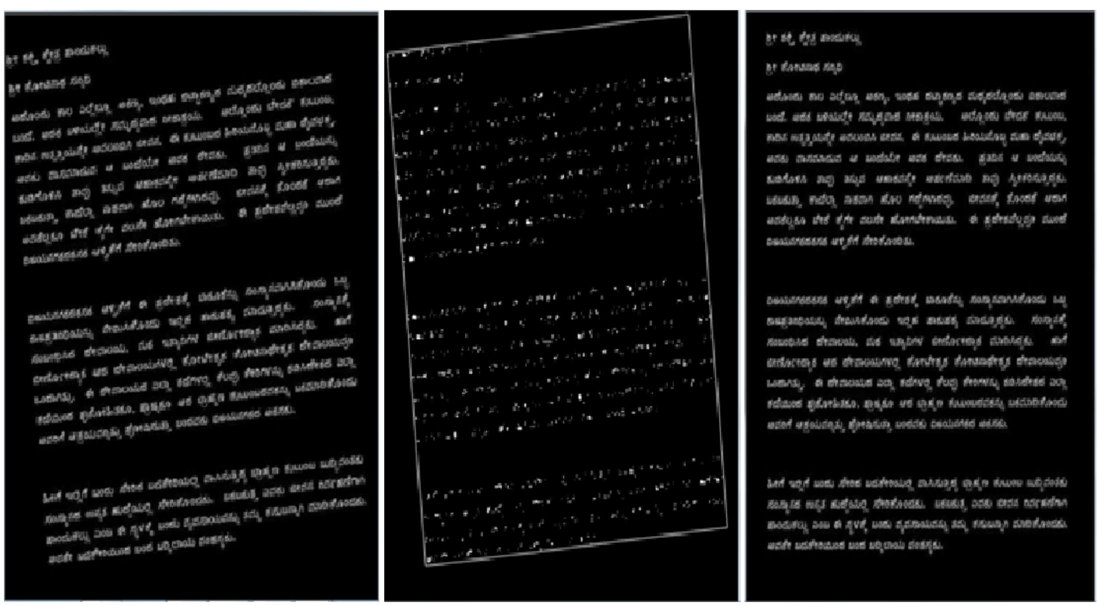
Authors	Method	Script	Range of Skew angle detected
Manjunath Aradhya [11]	Wavelets	Kannada	3,5,10,15
D. S. Guru, et. al., [10]	Connected component blotching and linear regression	All scripts	$\pm 90^{o}$
B. B. Chaudhari, et. al., [9]	Headline detection	Bangla and Devanagari	$\pm 45^{o}$
P. Shivakumara, et. al., [12]	Static and dynamic thresholding	English	0^{o}-30^{o}
Nandini N. [13]	Connected component analysis and Hough transform	English	0^{o}-20^{o}
Proposed	Bounding Box	Kannada	$\pm 45^{o}$

Table 2. Comparison of Proposed Method with the Existing Methods for Line Segmentation

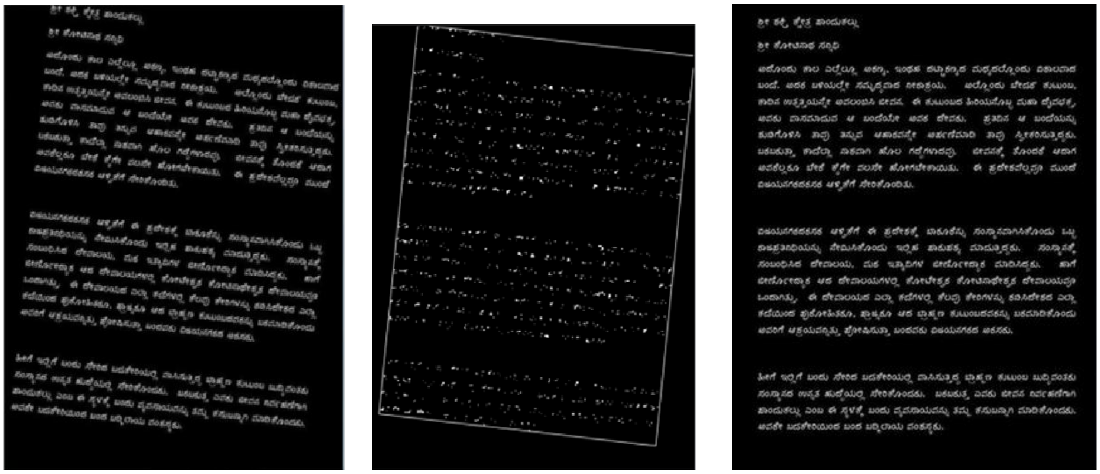
Author	Segmentation Method	Size of dataset	Segmentation rate
Alireza Alaei, et. al., [37]	Potential Piece-wise Separation Line technique	204	94.98%
Alireza Alaei, et. al., [37]	Stripe based approach	204	95.32%
M. Aradhya, et. al., [40]	Component extension technique	250	Not specified
Proposed	Hough Transform	40	91%



(a) (b) (c)

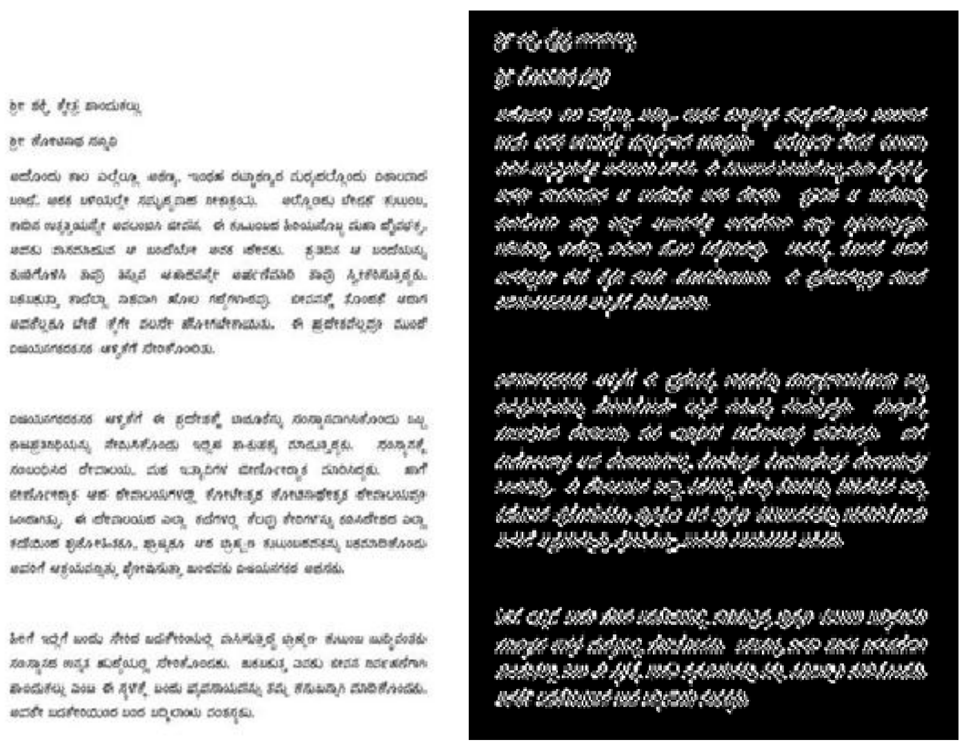


(a) (b) (c)

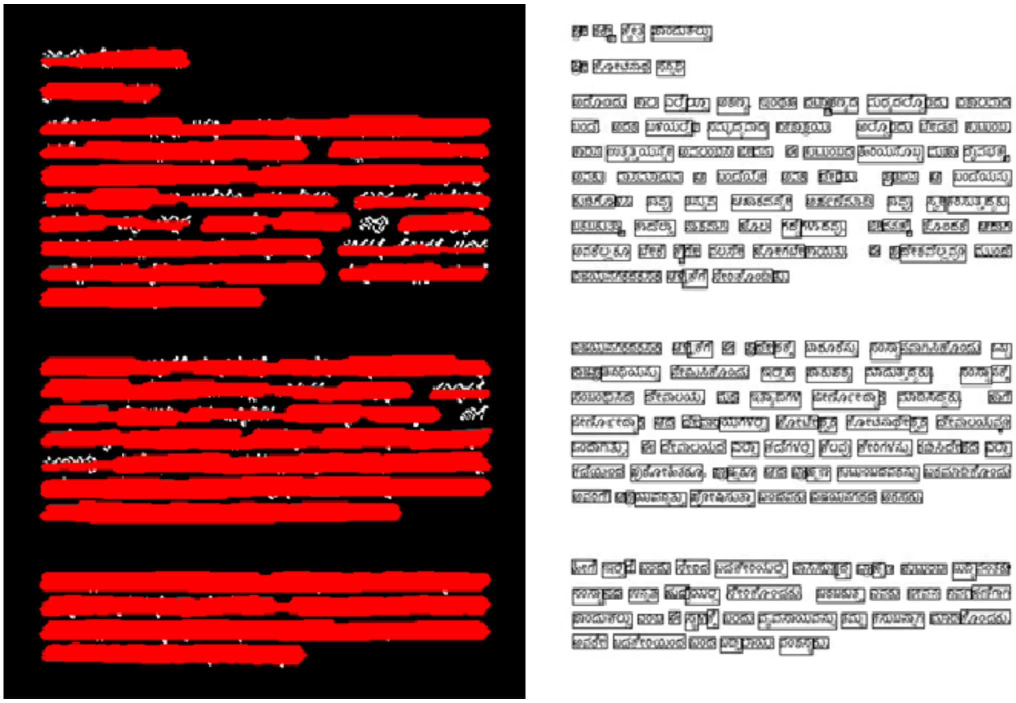


(a) (b) (c)

Figure 5. Results of Skew Detection and Correction (a) Binarised original document(skewed) (b) Bounding box on the document (c) Deskewed document with the estimated skew angle



(a) (b)



(c) (d)

Figure 6. Results of Segmentation (a) original document (b)binarised document (c) text line segmentation (d) word segmentation

Acknowledgements

Authors would like to thank Ms Varsha V, Ms Varna S and Ms Vraushali of the Department of Information Science and Engineering, P E S Institute of Technology, Bangalore, India who has helped us for collecting the documents, preparing the ground truths and their supporting help in conducting the experiments. The authors would also like to thank all writers who contributed for this dataset.

References

[1] K. S. Murthy, G. H. Kumar, P. Shivakumar, P. R. Ranganath, "Nearest Neighbour Clustering approach for line and character segmentation in epigraphical scripts", Proceedings of International Conference on Cognitive Systems (ICCS-2004), New Delhi, **(2004)** December 14-15.

[2] H. K. Kwag, S. H. Kim, S. H. Jeony and G. S. Lee, "Efficient skew estimation and correction algorithm for document images", Image and vision Computing, vol. 20, **(2002)**, pp. 25-35.

[3] H. S. Baird, "The Skew Angle of Printed Documents", Proc. Society of Photographic Scientific Eng., vol. 40, **(1987)**, pp. 21-24.

[4] PostlW, "Detection of linear oblique structure and skew scan in digitized documents", In Proc. Int.Conf. on Pattern Recognition, **(1986)**, pp. 687-689.

[5] A. Hashizume, P. S. Yeh and A. Rosenfeld, "A method of detecting the orientation of aligned components", Pattern Recognition Letters, vol. 4, **(1986)**, pp. 125-132.

[6] H. Yan, "Skew correction of document images using interline cross-correlation", Computer Vision, Graphics, and Image Processing, vol. 55, **(1993)**, pp. 538-543.

[7] S. N. Srihari and Govindaraju, "Analysis of textual images using the Hough transform", Machine Vision Appl., vol. 2, **(1989)**, pp. 141-153.

[8] P. Shivakumara, G. H. Kumar, D. S. Guru and P. Nagabhushan, "A novel technique for estimation of skew in binary text document images based on linear regression analysis", Sadhana, vol. 30, Part 1, **(2005)** February**,** pp. 69-85.
[9] B. B. Chaudhari and U. Pal, "Skew angle detection of digitized Indian script document", PAMI, vol. 19, no. 2, **(1997)**, pp. 182-186.
[10] D. S. Guru, P. Punitha and S. Mahesh, "Skew Estimation in Digitized Documents: A Novel Approach", ICVGIP, **(2004).**
[11] V. N. M. Aradhya, "Document Skew Estimation: An Approach based on Wavelets", ICCCS'11**, (2011),** pp. 359-364.
[12] P. Shivakumara, G. H. Kumar, D. S. Guru and P. Nagabhushan, "Skew Estimation of Binary Document Images Using Static and Dynamic Thresholds", Proceedings: National Workshop on IT Services and Applications (WITSA2003), **(2003)** February 27-28**,** pp. 1-5.
[13] N. Nandini, S. Murthy and G. H. Kumar, "Estimation of Skew Angle in Binary Document Images Using Hough Transform", World Academy of Science, Engineering and Technology, vol. 42, **(2008)**, pp. 44-49.
[14] B. Yuan, L. K. Kwoh and C. L. Tan, "Finding the best-fit bounding-boxes", proceedings of the 7th international conference on Document Analysis Systems, **(2006)**, pp. 268-279.
[15] B. Yu, A. K. Jain, "A robust and fast skew detection algorithm for generic documents", Pattern Recognition, vol. 29, no. 10, **(1996),** pp. 1599-1629.
[16] K. S. S. Kumar, A. M. Namboodiri and C. V. Jawahar, "Learning Segmentation of Documents with Complex Scripts", ICVGIP 2006, LNCS 4338, **(2006)**, pp. 749–760.
[17] Z. Razak, K. Zulkiflee, M. Y. I. Idris, E. M. Tamil, M. Noorzaily, M. Noor, R. Salleh, M. Yaakob, Z. M. Yusof and M. Yaacob, "Off-line Handwriting Text Line Segmentation: A Review", IJCSNS International Journal of Computer Science and Network Security, vol. 8, no. 7, **(2008)** July, pp. 12-20.
[18] M. Arivazhagan, H. Srinivasan and S. N. Srihari, "A Statistical Approach to Handwritten Line Segmentation", Proceedings of SPIE Document Recognition and Retrieval XIV, San Jose, CA, **(2007)** February.
[19] Y. Li, Y. Zheng, D. Doermann and S. Jaeger, "A new algorithm for detecting text line in handwritten documents", International Workshop on Frontiers in Handwriting Recognition, **(2006)**, pp. 35–40.
[20] 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, **(1994**), pp. 117-135.
[21] L. Likforman-Sulem, A. Hanimyan and C. Faure, "A Hough based algorithm for extracting text lines in handwritten documents", Third International Conference on Document Analysis and Recognition, vol. 2, **(1995)** August**,** pp. 774-777.
[22] C. Weliwitage, A. L. Harvey and A. B. Jennings, "Handwritten Document Offline Text Line Segmentation", Proceedings of Digital Imaging Computing: Techniques and Applications, **(2005)**, pp. 184-187.
[23] D. Sarkar and R. Ghose, "A bottom up approach of line segmentation from handwritten text".
[24] U. pal P. P. Roy and J. Liados, "Morphology based handwritten line segmentation using foreground and backgroud information", In the proc. of Int'l conference on Frontiers in Handwriting Recognition, **(2010).**
[25] L. Kaur, N. K. Garg and M. K. Jindal, "A new method for line segmentation of handwritten hindi text", In the proceeding of 7th Intl conference on Information technology, **(2010)**, pp. 392–397**.**
[26] M. kundu M. Nasipuri S. Basu, C. Chaudhuri and D. K. Basu, "Text line extraction from multi-skewed handwritten documents", Patten Recognition, vol. 40, **(2007)**, pp. 1825–1839.
[27] G. Louloudis, B. Gatos, I. Pratikakis and C. Halatsis, "Line And Word Segmentation of Handwritten Documents", Journal Pattern Recognition archive Volume 42 Issue 12, December, **(2009)**, pp 3169-3183.
[28] F. Luthy, T. Varga and H. Bunke, "Using Hidden Markov Models as a Tool for Handwritten Text Line Segmentation", Ninth International Conference on Document Analysis and Recognition, Curitiba, Brazil, **(2007)**, pp. 8-12.
[29] V. K. Koppula and A. Negi, "Using Fringe Maps for Text Line Segmentation in Printed or Handwritten Document Images", 2010 Second Vaagdevi International Conference on Information Technology for Real World Problems, **(2010)**, pp. 83-88.
[30] N. K. Garg, L. Kaur and M. K. Jindal, "A New Method for Line Segmentation of Handwritten Hindi Text", Proceedings 2010 Seventh International Conference on Information Technology, **(2010)**, pp. 392-397.
[31] R. Kumar and A. Singh, "Detection and Segmentation of Lines and Words in Gurmukhi Handwritten Text", 2010 IEEE 2nd International Advance Computing Conference, **(2010)**, pp. 353-356.
[32] G. Louloudis, N. Stamatopoulos and B. Gatos, A Novel Two Stage Evaluation Methodology for Word Segmentation Techniques, 2009 10th Int'l Conf. on Document Analysis and Recognition, **(2009)**, pp. 686-690.
[33] http://en.wikipedia.org/wiki/Sobel_operator.
[34] http://en.wikipedia.org/wiki/Hough_transform.
[35] N. Nandini, S. Murthy K and G. H. Kumar, "Estimation of Skew Angle in Binary Document Images Using Hough Transform", World Academy of Science, Engineering and Technology, vol. 42, **(2008)**, pp. 44-49.

[36] J. Illingworth and J. Kittler, "A Survey Of The Hough Transform", Computer Vision, Graphics, and Image Processing, vol. 44, **(1988)**, pp. 87-116.
[37] A. Alaei, P. Nagabhushan and U. Pal, "A Benchmark Kannada Handwritten Document Dataset and its Segmentation", International Conference on Document Analysis and Recognition, **(2011)**, pp. 141-145.
[38] B. Gatos, N. Stamatopoulos and G. Louloudis, ICDAR 2009 Handwriting Segmentation Contest, Proc. of 10th ICDAR, **(2009)**, pp. 1393–1397.
[39] A. Alaei, U. Pal and P. Nagabhushan, "A new scheme for unconstrained handwritten text-line segmentation", Pattern Recognition, vol. 44, no. 4, **(2011)**, pp. 917–928.
[40] V. N. Manjunath Aradhya and C. Naveena, "Text Line Segmentation of Unconstrained Handwritten Kannada Script", ICCCS'11, **(2011)**, pp. 231-234.

Authors

Mamatha H. R.

She received her B E degree in Computer Science and Engineering from the Kuvempu University in 1998.and the M.Tech degree in Computer Networks and Engineering from the Visvesvaraya Technological University in 2006.Since 2008 she has been a Ph.D. student at the Visvesvaraya Technological University. She has 14 years of teaching experience. Currently she is working as an Assistant Professor in the Department of Information Science and Engineering, P E S Institute of Technology. Her current research interests include pattern recognition and image processing. She has published 14 papers in various Journals and Conferences of International repute. She is a life member of Indian Society for Technical Education. She has mentored students for various competitions at international level.

.

Dr. Srikanta Murthy K.

He received B.E (Electrical & Electronics Engineering). degree in 1986, M.Tech (Power Systems) in 1996 from National Institute of Engineering, University of Mysore and Ph.D. degree in Computer Science from University of Mysore in 2006. He joined the faculty of Electrical Engineering in NIE in 1987 and worked for K V G College of Engineering at various positions from 1991-2004. Presently he is working as Professor and Head, Department of CS&E, P E S School of Engineering, Bangalore, India. He has served as Chairman Board of Examiners in computer science in Mangalore University and also the member of local inspection committee, Mangalore University and Visvesvaraya Technological University. Currently he is guiding 5 candidates towards the doctoral degree and also guided many projects at PG level. His current research interests are computer vision, image processing and pattern recognition. He has published 60+ papers in various Journals and Conferences of International repute. He is a life member of Indian Society for Technical Education, Indian Society of Remote Sensing, Computer Society of India, Society of Statistics and Computer Applications and associate member of Institute of Engineers.