1. **INTRODUCTION**
   1. **PROBLEM STATEMENT**

To be able to extract the Bengali texts from images

* 1. **PROBLEM DEFINITION**

To develop an algorithm to extract Bengali texts from images

* 1. **OBJECTIVE**

To develop an effective algorithm to extract Bengali texts from images .The text extracted can then have multiple uses. For example Google’s Image Translator where the text extracted from the image is translated to the language desired by the user.

* 1. **LITERATURE SURVEY**

[1]A novel text line extraction technique is presented for multi-skewed document images of handwritten English or Bengali text. It assumes that hypothetical water flows, from both left and right sides of the image frame, face obstruction from characters of text lines. The stripes of areas left unwetted on the image frame are finally labelled for extraction of text lines. The success rate of the technique, as observed experimentally, are 90.34% and 91.44% for handwritten Bengali and English document images, respectively. The work may contribute significantly for the development of applications related to optical character recognition of Bengali/English text.

[2] Text segmentation is an inherent part of an OCR system irrespective of the domain of application of it. The OCR system contains a segmentation module where the text lines, words and ultimately the characters must be segmented properly for its successful recognition. The present work implements a Hough transform based technique for line and word segmentation from digitized images. The proposed technique is applied not only on the document image dataset but also on dataset for business card reader system and license plate recognition system. For standardization of the performance of the system the technique is also applied on public domain dataset published in the website by CMATER, Jadavpur University. The document images consist of multi-script printed and hand written text lines with variety in script and line spacing in single document image. The technique performs quite satisfactorily when applied on mobile camera captured business card images with low resolution. The usefulness of the technique is verified by applying it in a commercial project for localization of license plate of vehicles from surveillance camera images by the process of segmentation itself. The accuracy of the technique for word segmentation, as verified experimentally, is 85.7% for document images, 94.6% for business card images and 88% for surveillance camera images.

**1.5 BRIEF DISCUSSION ON PROBLEM**

Text segmentation is an inherent part of an OCR system irrespective of the domain of application of it. The OCR system contains a segmentation module where the text lines, words and ultimately the characters must be segmented properly for its successful recognition.

We mainly work this project on the fact that the matra on most of the Bengali words form a line which can be identified using hough’s transform.

1. **CONCEPTS AND PROBLEM ANALYSIS**

**2.1 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 line 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 ob-jects. 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 equa-tion. 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 w.r.t. x axis and *c* is the intercept on y axis made by the line. Any line can be cha-racterized 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

*c* = *y − mx*

The set of (*m, c*) values corresponding to the lines pass-ing through point (*x, y*) form a line in (*m, c*) space. Every point in image space (*x, y*) corresponds to a line in para-meter space (*m, c*) and in the reverse way, each point in (*m, c*) space corresponds to a line in image space (*x, y*).

The Hough transform works by letting each feature point (*x, y*) vote in (*m, c*) space for each possible line pass-ing through it. These votes are totalled in an accumulator.

Suppose that a particular (*m, c*) has one vote—this means that there is a feature point through which this line passes. If it has two votes then it means that two feature points lie on that line. If a position (*m, c*) in the accumula-tor has *n* votes, this means that *n* feature points lie on that line.

**The Hough Transform Algorithm**

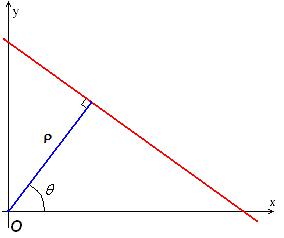
1. Find all of the desired feature points in the image space.
2. For each feature point in image space
3. For each possibility *i* in the accumulator that passes through the feature point
4. Increment that position in the accumulator
5. Find local maxima in the accumulator.
6. On requirement map each maxima in the accumulator back to image space

**Alternative representation of Lines**

The slope-intercept form of a line as discussed above-has a problem with vertical lines: both *m* and *c* are infi-nite. To eliminate this problem of representing the point in the (*m, c*) space another way of expressing a line is in (*ρ,* *θ*) form is used as

*x* cos *θ* + *y* sin *θ* = *ρ*

One way of interpreting this is to drop a perpendicular from the origin to the line. *θ* is the angle that the perpen-dicular makes with the *x*-axis and *ρ* is the length of the perpendicular. *θ* is bounded by [0*,* 2*π*] and *ρ* is bounded by the diagonal of the image. Instead of making lines in the accumulator, each feature point votes for a sinusoid of points in the accumulator. Where these sinusoids cross,

****

there are higher accumulator values. Finding maxima in the accumulator still equates to finding the lines.

The ( ρ*,* *θ*) plane is sometimes referred to as *Hough* *space* for the set of straight lines in a two dimensional im-age space. The steps of implementation can be summa-rized below.

* For each image data point, a number of lines are

plotted going through it, all at different angles.

 For each line a line is plotted which

is perpendicular to it and which intersects

the origin.

* The length and angle of each dashed line is meas-ured.
* These the steps are repeated for each data point.
* A graph of length against angle, known as a Hough space graph, is then created.

For the line-matching Hough Transform, the orienta-tion of the line is one of the parameters. If the orientation parameter is not used then matches in a specific orienta-tion can be found out. The orientation parameter can be changed sequentially in an incremental way to find out all the lines oriented in different directions.

**2.2 WORK FLOW**

* Image Acquisition
* Image Preprocessing
* Edge Detection
* Hough Transform for line extraction
* Connected Component Labelling Algorithm
* Hough Transform for word extraction
* Bounding Box creation

**2.3 IMAGE ACQUISITION**

Image acquisition is done through scanner or mobile camea or CCTV camera





**2.4 IMAGE PREPROCESSING**

The images for all the domains of application suffer from slight rotation due to the process of capturing itself. A rotation algorithm is applied to make them horizontally aligned. The color images are then converted to grey level images by finding the grey value of each pixel loated at *(i,j)* from the 24-bit color value of it using the followingformula:

*grey*(*i*, *j*) 0.59 *red*(*i*, *j*) 0.30 *green*(*i*, *j*) 0.11 *blue*(*i*, *j*)



**2.5 EDGE DETECTION**

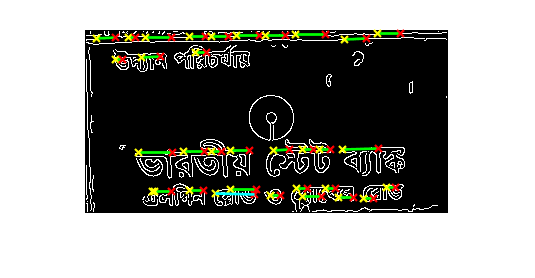
The edge map of the grey image is generated using So-bel’s edge operator [5]. Mathematically, the operator uses four 3×3 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal direction, one for vertical direction and two for the two diagonal directions. All four masks (horizontal vertical and two diagonal) are used for the detection of edge gradient. The edge gra-dient is then binarised using any binarization algorithm to geneate the binarized edge map of the image.



**2.6 HOUGH TRANSFORM FOR LINE EXTRACTION**

Hough transform is applied on the binarized edge map to generate the Hough image of it. For this purpose, the parameters of the Hough transform, like delta Ro, delta Theta, start Theta, end Theta, connected Distance and pixels-Count are initialized or tuned in such a way that the lines are extracted as a set of connected words. In the current work, any pixel-level line having a skew angle of 850 to 950 is considered with delta Theta taken as 10. The connected Distance and pixels Count values are kept as 50 and 30 respectively. The Hough image is stored as bmp file for analysis of the performance of the system.

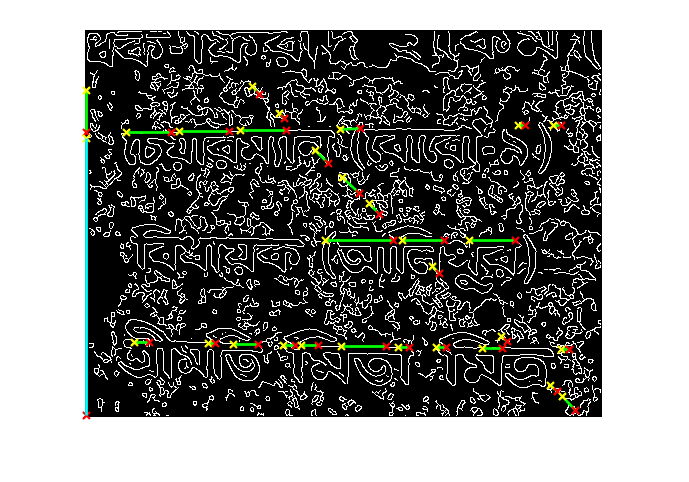




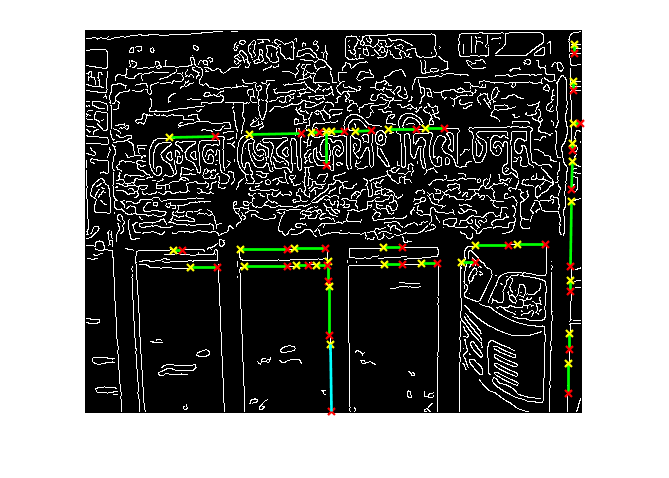
















**2.7 CONNECTED COMPONENT LABELLING ALGORITHM**

All the marked white band of lines in tho Hough image are segmented through CCL algorithm. In this algorithm, any white pixel searches for its white neighbours. The neighbours will also search for their white neighbours in a recursive way. In the present work, 4-connected neighbours are searched and non-recursive function call is used to reduce usage of system resource and time complexity. Each and every connected component is labelled accordingly for future use in word segmentation section.

**2.8 HOUGH TRANSFORM FOR WORD EXTRACTION**

Hough transform is applied on the line segments to gen-erate the Hough image of it at the words level. For this purpose, the parameters of the Hough transform, like deltaRo, deltaTheta, startTheta, endTheta, connectDis-tance and pixelsCount are initialized or tuned in such a way that the words are extracted as a set of connected characters. In the proposed work, any pixel-level line hav-ing a skew angle of 300 to 1200 is considered with delta-Theta taken as 10. The connectDistance and pixelsCount values are kept as 20 and 2 respectively. The new Hough image is stored as bmp file for analysis of the perfor-mance of the system.

The CCL algorithm as described in section 2.7 is again applied on the Hough image to segment it terms of words.

**2.9 BOUNDING BOX CREATION**

The location of the word segments as generated during the time of running CCL algorithm are stored in terms of starting row, starting column, ending row and ending column of the component. The locations of the segments are marked in terms of bounding box drawn over the bi-nary image. The boxed binary image is then stored as a bmp file as an outcome of the system as well as for evalu-ation of the performance of it.

**3. CONCLUSION**

Text segmentation is an inherent part of an OCR system irrespective of the domain of application of it. The OCR system contains a segmentation module where the text lines, words and ultimately the characters must be segmented properly for its successful recognition.

It is also important in image translators for the very same reason , i.e. the t ext lines, words and ultimately the characters must be segmented properly for its successful recognition.

This project aims to develop an algorithm to efficiently extract the text from the images provided. In essence, we have used the hough’s transform algorithm to detect the lines from the greyscale image of the provided rgb image.

After getting the matra ,i.e. the line, we get connected components connected to the line and label them. We then box them out to segment them which can then be feeded to the OCR or translator.

**REFERENCES**

1. [S.Basu](https://www.sciencedirect.com/science/article/abs/pii/S0031320306004213#!) , [C.Chaudhuri](https://www.sciencedirect.com/science/article/abs/pii/S0031320306004213#!), [M.Kundu](https://www.sciencedirect.com/science/article/abs/pii/S0031320306004213#!) , [M.Nasipuri](https://www.sciencedirect.com/science/article/abs/pii/S0031320306004213#!) [D.K.Basu](https://www.sciencedirect.com/science/article/abs/pii/S0031320306004213#!) . Text line extraction from multi-skewed handwritten documents
2. Satadal Saha, Subhadip Basu, Mita Nasipuri and Dipak Kr. Basu . A Hough Transform based Technique for Text Segmentation
3. Digital Image Processing by Rafael C. Gonzalez and Richard Eugene Woods