© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 14 Issue: 04 | July - September 2023
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

A Technique is developed to over-segmentation of the handwritten word images acquisition:

Dr. Anoop Sharma,
Dept. of computer & Research Center,
Singhania University pacheri Bari Raj.
Ms. Rajnesh Kumari,
Research Scholar, Dept. of computer & Research Center,
Singhania University pacheri Bari Raj.

Abstract: Off-Line handwriting segmentation and recognition has been a challenging and exciting area of research for many years. The popularity of this field of research is mainly due to the unconstrained and cursive nature of human handwriting. The segmentation and recognition of such type of handwritten script is still an open problem and is an active area of research these days. The character recognition accuracy of an OCR system can be improved remarkably if the characters within a word are correctly isolated. Hence, segmentation is the most crucial step in the off-line cursive handwritten script recognition process. Good segmentation results are always welcome.

Key Words: Handwriting, Character, Segmentation, Proposed.

Introduction: The selection of a segmentation technique depends on the nature of the script to the segmented. The proposed segmentation technique is proposed for segmentation of touched characters from the handwritten words of varying length written on a noisy background. This new segmentation technique in which the segmentation points are located after thinning the word image to get the stroke width of the single pixel. The knowledge of shape and geometry of English characters is used in the segmentation algorithm to detect ligatures. The proposed segmentation approach is tested on a local database and high segmentation accuracy is found to be achieved.

Objectives: implementing effective offline feature Extraction technique.

Result & Discussion:

Preparation of Handwritten Words Local Database:

For conducting the segmentation experiment by the proposed segmentation technique, handwriting samples from 10 different people (age 15-50 years) has been gathered. Some of these samples are written on white paper and others on a colored or a noisy background. Exactly 200 words have been selected randomly from these handwriting samples containing all shapes of English characters written by those persons. Some of the word image samples from the collected database are shown in fig. 1.1

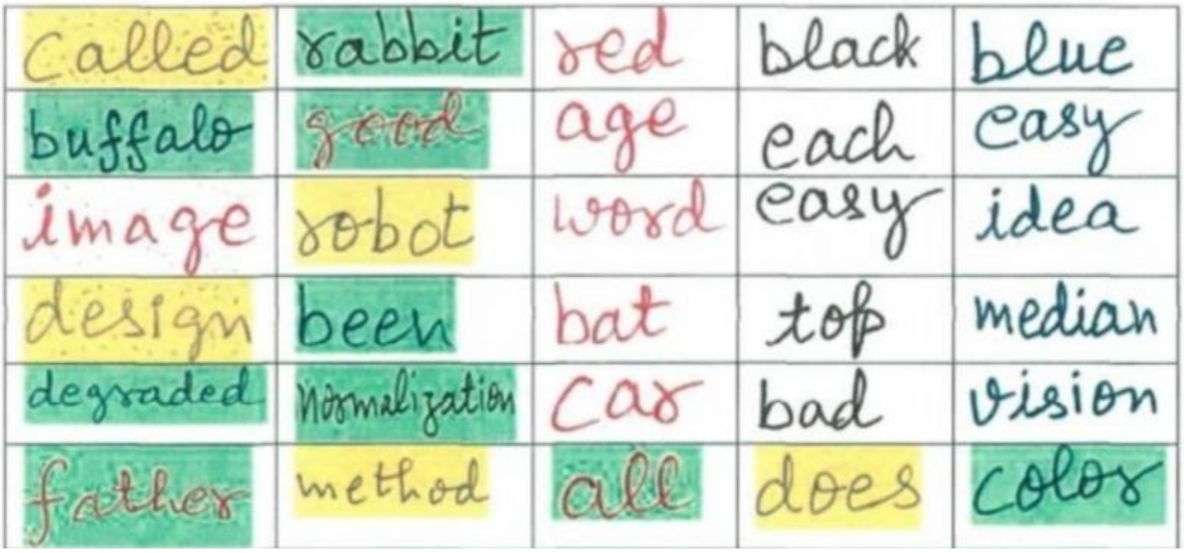


Fig. 1.1 Handwritten Word Images Samples having Touched Characters

Handwritten Word Image Acquisition

In image acquisition, the word images are acquired through a scanner or a digital camera. The input word images are saved in JPEG or BMP formats for further processing. Three such handwritten word images from the local database are shown in fig. 1.2

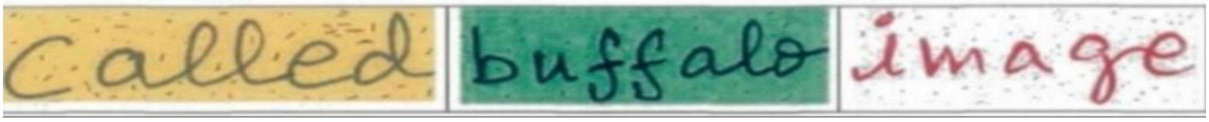


Fig. 1.2 Handwritten Word Images Samples having Touched Characters

Word Image Preprocessing

The aim of preprocessing is to eliminate the inconsistency that is inherent in cursive handwritten words. The handwriting samples may be written on a noisy or colored background and also the quality of the word images may be degraded due to the noise that is introduced in the process of scanning or capturing the word images. It is necessary to remove the background noise to improve the quality of the word images for the segmentation experiment. Preprocessing of the word image is of main concern so that the segmentation of the characters from the word images may be carried out correctly. After the segmentation process, it is also very important to deal with the preprocessing of each segmented character to improve accuracy of the recognition process.

The various preprocessing techniques that have been employed in an attempt to increase the performance of the segmentation process are as follows:

Thresholding

In this phase of preprocessing, the RGB images in BMP format shown in fig. are converted to grayscale format by retaining the illumination while eliminating the hue and saturation as shown in fig. This preprocessing step is necessary so as to overcome the problems that may arise due to the use of pens of different colors and different intensities on various noisy and colored backgrounds.

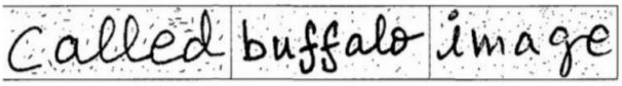
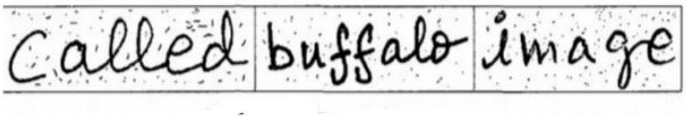


Fig. 1.3 Word image in Grayscale Format

The grayscale images are then converted in a binary matrix format. The resultant binary images have values of 0 each for all the foreground black pixels and 1 each for all the background white pixels. The threshold parameter is so chosen that some negligible information of the characters is lost. The resulting word images after background noise removal and binarization using gray scale intensity threshold are shown in fig. It is advantageous to store these handwritten word images in this bilevel image format because it is easy to handle only two levels of colors. Also, these images take less storage and are computationally less expensive. Hence, further processing becomes fast.

Fig. 1.4 Word Images in Binary Form



Thinning and Skeletonization

Skeletonization is a process in which the foreground region in a binary image is reduced to a skeletal remnant. During the process, the connectivity of the original region is preserved while removing a maximum number of original foreground pixels. Thinning is an image morphological operation in which selected foreground pixels are removed by eroding an image until it becomes one-pixel wide. It produces a skeleton of the object present in the image and makes it easier to recognize the object as character. Thinning process is usually applied to a binary image and the output is also another binary image. The process of thinning erodes an object over and again, without breaking it, until the width remains to one-pixel wide.

A large amount of variability may be present among the handwritten words because writers can use different type of pens of unequal stroke width while giving their handwriting samples. The thickness that varied from one word to another must be uniform. The thinning process delivers all the words used in the propose experiment, a uniform stroke width of one-pixel.

Although thinning the word image is disadvantageous rather than beneficial because sometimes a huge amount of important information is lost e.g. some filled holes in the word images may get disappeared after thinning the word images. However, it is necessary to make the stroke width uniform for every handwritten word images so as to make the thickness of each word to one-pixel wide. Three such handwritten word images after thinning are shown in fig. 1.5

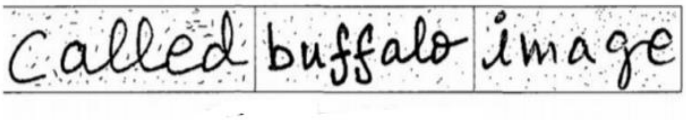


Fig. 1.5 Word Image after Thinning

Noise Removal

Noise (small dots or foreground components) may be introduced easily into an image while scanning the handwritten word image acquisition. It is very necessary to eliminate the noise from the word images so as to make the word images fit for further processing. MATLAB's 'bwareaopen' method is used to morphologically open the binary image by removing small objects that have less than a particular number (user pixel) image by removing small objects that have less than a particular number (user specified) of pixels and producing another binary image. The small noise dots are removed by using 'bwareaopen' but some small portions of the characters e.g. dot '.' Of character 'i, j' are also lost.

The methods ‘bwlabel’ and ‘regionprops’ of MATLAB are used to highlight the pixels that are removed as shown in fig. 1.6

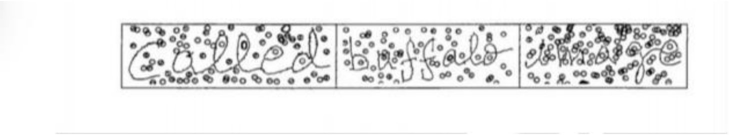


Fig. 1.6 Noise Detection in Word Images

A logical AND operation of the dilated characters with the pixels removed by ‘bwareaopen’ is performed. The portions of the characters pixels which are very near to the character image are put back. The resultant image without noise dots while retaining the portions of the characters is shown in fig. 1.7

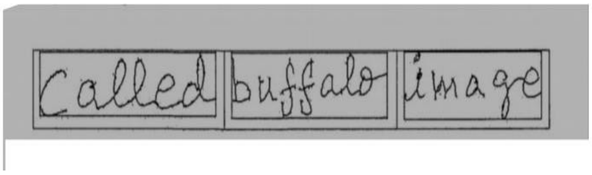


Fig. 1.7 Word Image after Noise Removal

Cropping of Handwritten Word Images

Image cropping is a process in which the extra space around the handwritten word image is removed. The outcome of ‘imcrop’ is a rectangular region with minimum area but containing the complete word image. The final word images after cropping operation are shown in fig. 1.8

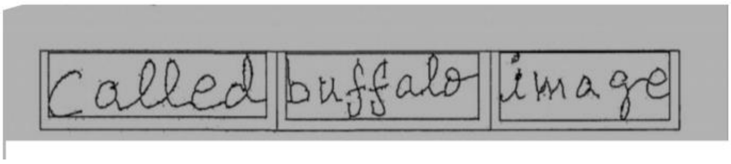


Fig. 1.8 Cropped Word Images

Segmentation Technique

Many segmentation techniques have been developed by the researchers in the recent years. These techniques are basically script dependent and may not work well if applied for segmentation of words written in any other script. For example, the technique developed for segmenting touched characters in Roman script may not work well to segment touched characters of a word written in Arabic or Chinese script.

Overview

There are two types of characters in English language. First type of characters are called “Closed Characters” and contain a loop or a semi-loop such as ‘a’, ‘b’, ‘c’, ‘d’, ‘e’, ‘g’, ‘o’, ‘p’, ‘s’ etc. Second type of characters are termed as “Open Characters” and are without a loop or a semi-loop e.g. ‘u’, ‘v’, ‘w’, ‘n’, ‘I’ etc. In case of open characters, it is very difficult to differentiate between ligatures and characters because of the cursive nature of handwriting. In case of cursive handwritten words, a ligature is a link (small foreground component) which is present between two successive characters to join them. Two consecutive ‘I’ characters may give an illusion of the presence of a character ‘u’ and vice versa. Two consecutive characters ‘n’ and ‘I’ may look like ‘m’. Also handwritten character ‘w’ may look like the presence of two consecutive characters ‘I’ and ‘v’. To overcome such type of challenges in the domain of cursive handwriting segmentation and recognition, a new segmentation approach is developed which is based on the analysis of the character's geometric features, such as, the shape of the character to identify the characters and the ligatures.

Research Methodology

After the preprocessing of the input handwritten word image, and width of the word image is calculated for the analysis of the ligatures in an accurate manner (Rehman et al., 2009). The word image is scanned vertically, from top to bottom, column wise and the number of foreground pixels in the inverted word image are counted in each column. The positions of all these columns are saved for which the sum of foreground black pixels is either 0 or 1. All these identified columns are termed as PSC (Potential Segmentation Columns).

The Problem of Over-segmentation

Many consecutive PSC are present in various groups in the whole word image where sum of foreground pixels are 0 or 1. This situation can be termed as over-segmentation. The over-segmentation problem is occurred in three cases. First, when the two consecutive characters in the word image are not touching each other and the sum of foreground pixels of the columns in this area are 0. Second, when the two consecutive characters in the word image are connected by a ligature are 1. Third, when the characters are Open Characters like 'u', 'w', 'm', 'n' etc. without having any loop or semi-loop. A ligature is present within all of these Open characters. Due to the presence of this ligature-with-in character, the characters of such type in the word image are still over-segmented and the sum of foreground pixels in these columns is also 1.

Solution of Over-Segmentation Problem

When there is a clear vertical space between two consecutive characters in a word image, the problem of over-segmentation is completely eliminated by taking average of all the PSC present in that area because the sum of foreground pixels for all these PSC is 0.

When there is ligature between two consecutive characters or there is a ligature within-character (open characters such as 'w', 'm' etc), the over-segmentation is eliminated to a great extent by taking average of those PSC which are at a distance less than a particular value (threshold) and by merging them into a single SC (Segmentation Column). The threshold value is the minimum distance (along the width of the word image) between consecutive PSCs and is so chosen that its value must be less than the width of the thinnest possible character (e.g. 'I', 'l') in a word image. By experimenting several times, the value of threshold is set to a value 7. This means that all those PSCs which are separated by a distance of 7 pixels or less by another PSC will be merged to a single SC (Segmentation Column).

Implementation

The steps followed during the implementation of the proposed segmentation technique are mentioned below:

Step 1: In the first step, the input word image is preprocessed by using various preprocessing techniques such as thresholding, binarization, thinning, noise removal and cropping. The collective outcome of all these preprocessing technique are summarized and reproduced in Fig. This preprocessed word image is taken as input image to be segmented into characters as shown in Fig.

Step 2: To minimize the computation complexity, the input word image is inverted for further processing. By complementing the input binary image, white pixels become the foreground pixels and the black pixels become the background pixels. Hence, it becomes easier to count the foreground white pixels represented by 1, in each column of the word image as show

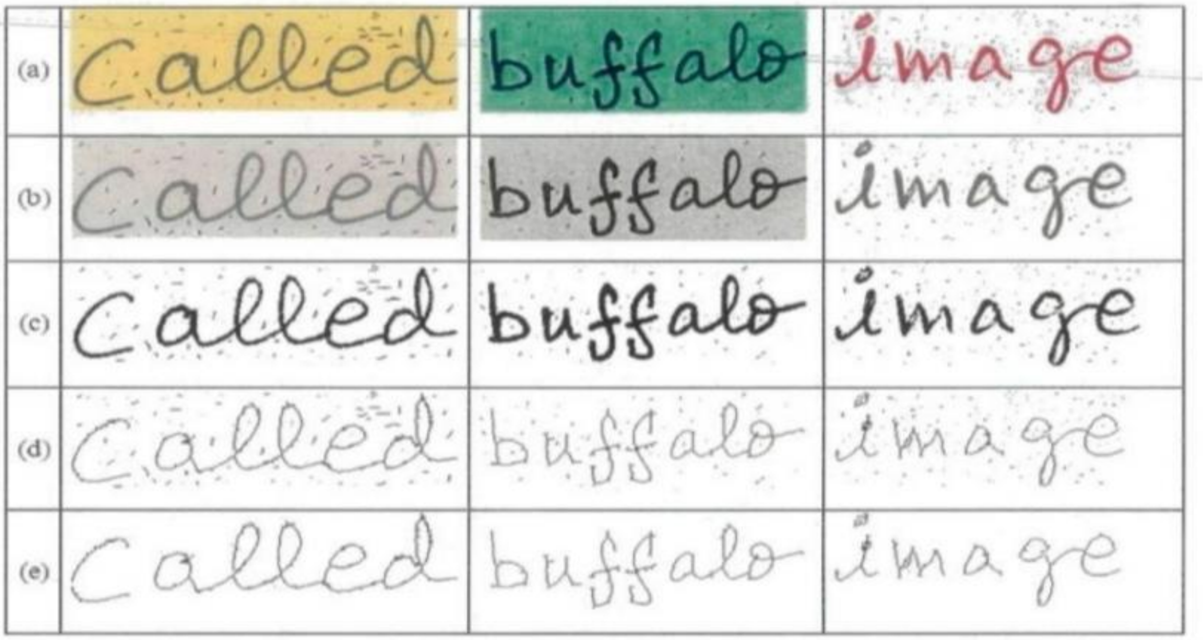


Fig. 1.9 Word Image preprocessing (a) Input Scanned Word Images; (b) Word Images after Gray Scale Intensity Threshold; (c) word Images in Binary Format; (d) Word images after thinning ; (e) Cropped Word Images after Noise Removal

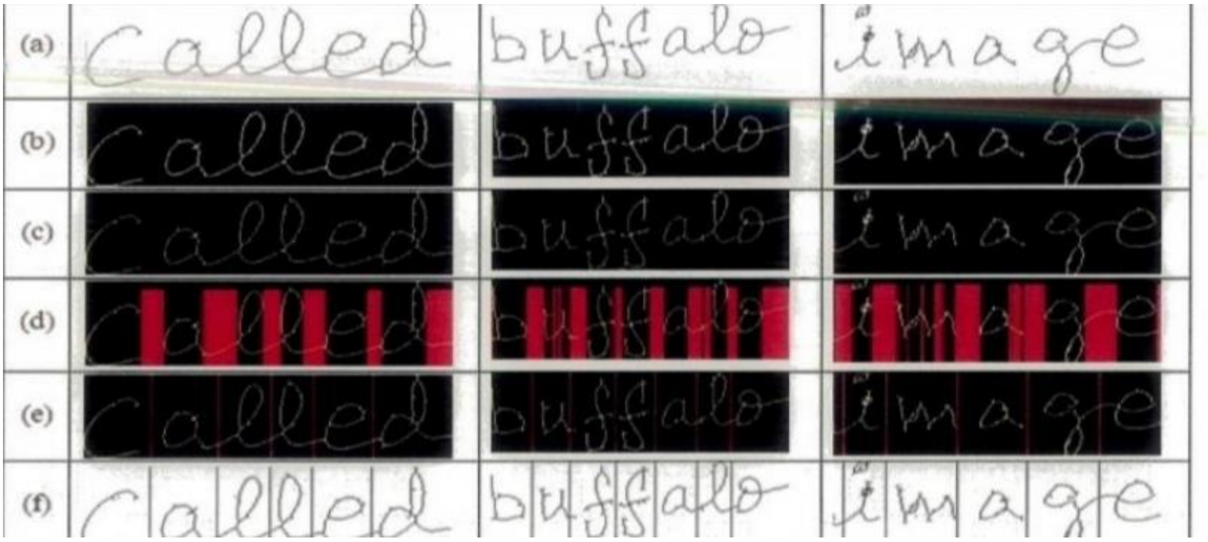


Fig. 1.10 Word Images Segmentation (a) Pre-processed word Images; (b) Inverted Binary Images; (c) RGB images; (d) Over-Segmentation in Images; (e) Image after removing Over-Segmentation; (f) Final Segmented Output Word Images

Step 3: This image is now converted from binary format to a RGB format as shown in fig. Now, it becomes computationally easier to display the PSC (Potential Segmentation Columns) in different color other than black and white.

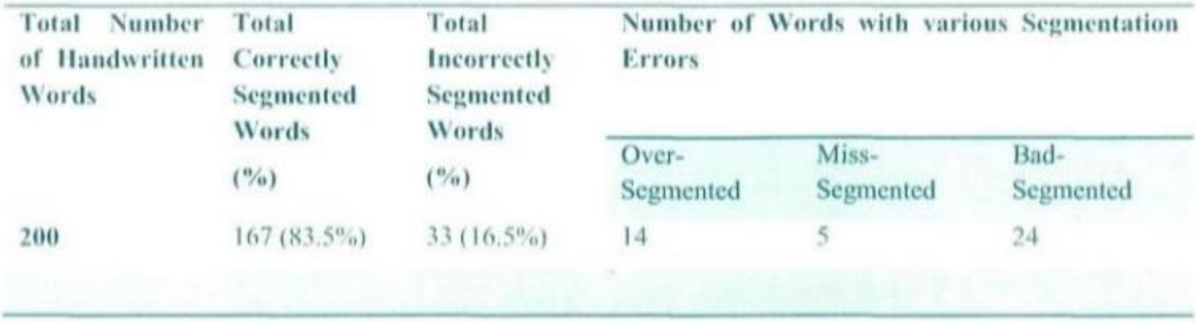
Step 4: All PSCs over-segmentation the word image are printed in red color as shown in fig. 4.18 ©. It is clear from fig 4.18(d) that each column in the word image, for which the sum of foreground white pixels is 0 or 1, is a PSC and vertically cuts the word image.

Step 5: All PSCs, which are at a distance less than a threshold value (7 pixels) from each other, are merged into a single column termed as Segmentation Column (SC) as shown in fig. 4.18 (e).

Step 6: The final segmented word image is obtained by changing the black background of the image with a white background as shown in fig.4.18 (f).

Result Analysis

For evaluation of the proposed segmentation approach, 200 handwritten word samples have been selected from the handwriting samples of 10 different writers. The performance of the proposed segmentation approach is judged on the basis of segmentation errors of the three types, namely, number of Over-Segmentations, Number of Miss-Segmentations and Number of Bad-Segmentations and is shown in Table 1.1.



Total Number of Handwritten Words	Total Correctly Segmented Words (%)	Total Incorrectly Segmented Words (%)	Number of Words with various Segmentation Errors		
			Over-Segmented	Miss-Segmented	Bad-Segmented
200	167 (83.5%)	33 (16.5%)	14	5	24

Out of 33 incorrectly segmented words, some words are over-segmented in one place as well as bad-segmented in some other place. While putting the results in Table 1.1, such types of words are counted in both of the error categories i.e. counted in over-segmented as well as bad-segmented. Similarly, in some words, the correct segmentation point is missed and shifted to some other place resulting in bad-segmentation. Some word images which are over-segmented or miss-segmented or bad-segmented are shown in fig. 1.11.

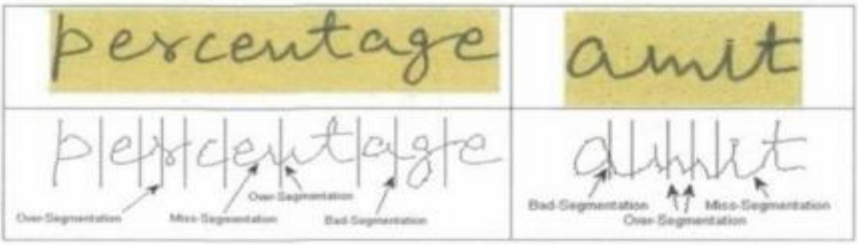


Fig. 1.11 Word Images showing All Type of Segmentation Errors

It is very difficult to compare the segmentation results achieved by the proposed approach with the segmentation results of some other researchers because different researchers used different databases of handwritten words and reported the segmentation results under various constraints, such as, some researchers assumed the absence of noise, some researchers collected the handwriting samples from different number of writers and so on. Although some researchers (Marti and Bunke, 2002; Hull,

1994) used various benchmark databases e.g. CEDAR or IAM for their experiment but they used different number of words from the benchmark database. As the character segmentation in word images is done before the character recognition phase, most of the researchers mentioned only in the recognition results and not the segmentation results.

Conclusion: A new vertical segmentation technique is developed to enhance the over-segmentation of the handwritten word image by thinning the word image to a single pixel width. The objective of the proposed approach is to over-segment the handwritten word image sufficient number of times to ensure that all possible character boundaries have been dissected. Another technique is also developed to merge than one successive segmentation points present between any two characters into a single segmentation point to enhance the segmentation performance.

References: Belaid (1997) OCR Print – *An overview. Chapter 2 in Survey of the state of the art in human language technology, 71-74* Gatos B, Louloudis G & Stamatopoulos N (2014) Segmentation of Historical Handwritten Documents into Text Zones and Text Lines *proceedings of International Conference on frontiers in Handwriting Recognition, ICFHR, 464-469.10..1109/ICFHR2014.84.*

Dutta K, Krishan P, Mathew M and Jawahar CV (2018) Offline Handwriting Recognition on Devanagari Using a New Benchmark Dataset. *13th IAPR International Workshop on Document Analysis Systems (DAS), Vienna 25-30.*

Gatos B, Louloudis G & Stamatpoulos N (2014) Segmentation of Historical Handwritten Documents into Text Zones and Text lines. *Proceedings of International Conference on Frontiers in Handwriting Recognition,* ICFHR, 464-469.10.1109/ICFHR2014.84.

Singh J and Lehal GS (2014*) Comparative Performance Analysis of feature (S)- Classifier Combination for Devanagari Optical Character Recognition System. International Journal of Advanced Computer Science and Applications, 5(6):37-42*