Classification of Hebrew Calligraphic Handwriting Styles: Preliminary Results

Itay Bar Yosef[1], Klara Kedem[1], Its'hak Dinstein[2],
Malachi Beit-Arie[3] and Edna Engel[3]

[1] Department of Computer Science, Ben Gurion University, Beer-Sheva, Israel

[2] Department of Electrical and Computer Enginering, Ben Gurion University, Beer-Sheva, Israel

[3] School of Library, Archive and Information Studies, The Hebrew University, Jerusalem, Israel

itaybar@cs.bgu.ac.il, klara@cs.bgu.ac.il , dinstein@ee.bgu.ac.il

Abstract

This paper presents preliminary results for document classification of ancient Hebrew manuscripts. The main goal is to analyze documents of different writing styles in order to identify the locations, the dates, and the writer of the test documents. This analysis depends crucially on good binarization of the corrupted manuscripts. We propose an accurate method for binarization of the manuscripts. We further propose and test topological features for handwriting style classification based a selected subset of the Hebrew alphabet. In our preliminary experiments we have used only two characters, the character Aleph and the character Lamed.

Our results so far yield 100% *correct classification of a set of fourteen documents written by fourteen different writers. .*

1. Introduction

Paleography is the study of ancient handwritten manuscripts. Among other things, it deals with dating and localizing of ancient and medieval scripts, and studying the development of the letters shape. Paleographical analysis of Hebrew manuscripts comprises of five essential operations whose goal is to establish concise paleographical identifications [1]:

(i) Applying an archeological approach to determine whether a manuscript is a single paleographical unit. Being a single unit means that the manuscript was not altered by additions to originally empty leaves, or by additions of missing parts.

(ii) Detecting whether one hand or more copied the manuscript. A document can be copied by more than one scribe and still be a single paleographical unit.

(iii) Establishing the paleographical type of the script. Medieval Hebrew scripts can be one of the following six entities: Ashkenazi, Italian, Sephardi, Byzantine, Oriental, and Yemenite.

(iv) Identifying the location where the manuscript was written both on codicological (technical features) and script grounds.

(v) Identifying the date when the document was written according to its codicological variables and style of script.

Figure 1 shows samples of two manuscripts.

The work reported in this paper is part of a project aimed to develop algorithms and tools for computerizing steps (iii)-(v) in the analysis of Hebrew manuscripts. Our algorithms are based on processing manuscripts according to their visual information. We apply and extend document and handwriting analysis techniques. Below we review some existing methods which are related to our work.

Two recent survey papers give excellent scientific background regarding document image analysis [2], and handwriting recognition [3]. A primary stage of most document image analysis tasks is document page segmentation [4], and document structure extraction [5]. The necessary preprocessing operations for off-line handwriting recognition are thresholding, noise filtering, and segmentation of lines, words, and characters [6]. Page segmentation is done in either *bottom-up* or *top-down* techniques. Bottom-up approaches [7] start with the classification of each pixel (or a neighborhood of pixels) as belonging to the background, or to a document object like text, graphics, photos, etc. Pixels belonging to text are grouped into characters, characters

Figure 1. Old Hebrew manuscripts examples

are grouped into words and lines, and so on. Top-down techniques [8] start with a page layout which is split into zones based on projection profiles. Once a document page is segmented, character recognition step is evoked, in which each connected component is taken as representing a character. Some special preprocessing may be required to separate touching characters or split characters from touching graphics [9].

An important step in any recognition application is *feature extraction*. A survey on feature extraction methods for character recognition can be found in [10]. The variety of possible extracted features is enormous, from features derived from binary images and characters contour, to features derived from skeleton representation of characters and from grayscale images. Features used for writer identification or for handwriting style analysis should represent the characteristics of the considered writers and styles. Offline cursive script word recognition techniques are surveyed in [11]. There is an extensive use of variations of the Hidden Markov Model (HMM) [12] in a wide range of cursive word recognition methods. In writer verification and identification applications [13], [14], [15], [16], test handwritings are compared with samples of handwriting from known sources, and the authorships are confirmed or unproved. In this sense, these applications are related to Paleographical research.

The first published work regarding the use of image processing for paleographical research (as far as we know) was published in 1971 [17]. Colette Sirat [18] is the author of another early publication reporting the use of computer image processing methods for paleographical research. Features based on run-length histograms were used in [19] for style classification of ancient Hebrew handwriting. An expert system using document analysis strategies for analysis and authentication of Hebrew manuscripts is reported in [20].

Since most of our analysis of the manuscripts is based on the characters shape (features based on binary image), a crucial step in our system is an accurate thresholding. In order to overcome the corrupted condition of many documents, we apply a multi-stage accurate binarization scheme. Then, we propose and test some topological features for classification. Our paper is organized as follows: the multi-stage binarization algorithm is presented in Section 2. The topological based features are introduced in Section 3. Section 4 presents the experimental results, and Section 5 concludes the paper and outlines our plans for continuing this research.

2. Binarization Method

2.1. Introduction

In general, historical document images are of poor quality because the documents have degraded over time due to their storage conditions, and the quality of the written parchment. As a result, the foreground and background are difficult to separate. The problem is particularly difficult because many of the document images have varying contrast, smudges, variable background intensity and the presence of seeping ink from the other side of the document. Due to the importance of characters shape and style in Paleographical analysis, the binarization method must be very accurate. We developed a *multi stage thresholding* method that gives excelent results both on dirty documents and on well preserved documents. Two examples of noisy text blocks are shown in Figure 2.

Due to the condition of the documents, a general global thresholding approach is not sufficient for separating the text and the background, since parts of the noise are darker then some parts of the characters. Several thresholding approaches have been reported in the literature on binarization of text documents with noisy background. In [21], Don presented a noise attribute thresholding method for document im-

Figure 2. (a) Two noisy text blocks.

age binarization. The method utilized noise attribute features based on a simple noise model to overcome the difficulty that some objects do not form prominent peaks in the histogram encountered by many conventional global thresholding methods. Negishi et al.[22] presented an automatic thresholding algorithm to extract the character bodies from the noisy background. They deal with extremely dirty and considerably large images, and cases where the gray levels of the character parts overlap with that of the background. Liu and Srihari [23] developed a thresholding algorithm based on texture features. Their proposed algorithm utilized two fundamental attributes of document images, in that, the characters normally occupy separable gray-level range in the gray-scale histogram and that the text images contain highly structured-stroke units. Tan et.al.[24] , established a method for removing of interfering strokes from double-sided handwritten documents based on the observation that the edges of the sipping strokes from the reverse side are not as sharp as those on the front side, they adopt an edge detection approach to suppress unwanted background patterns. In [25], the autors compared several algorithms for text binarization in difficult documents: Niblack's method [26], quadratic integral ratio (QIR) technique [27], Yanowitz and Bruckstein's method [28], and two new techniques proposed by the authors: The mean-gradient technique which is an improved version of Niblack's Method and a background subtraction technique based on graylevel morphology. In our documents, parts of the characters are faded due to the condition of the documents, a fact which makes edge based binarization methods unsuitable, yet the fundamental body of the characters is easier to be detected. This fact leaded us to adopt a region growing scheme in which we first detect the fundamental body of the characters, and then apply a growing process to grow the characters to their final form.

2.2. Multi-Stage Thresholding

Multi-stage thresholding can be viewed as a process of gradually reducing the search space of threshold candidate values stage by stage, where different information from the image is processed in each stage until the final threshold value is chosen. The threshold selection contains the following stages:

I. Global Thresholding.

II. Discarding irrelevant objects

III. Local component processing

IV. Postprocessing.

Next we describe each of the above steps.

I. Global thresholding. The objective of this stage is to narrow the search space of foreground candidates, and to produce spatial information on text lines and characters. This is achived by under-thresholding the image using a global thresholding method. We use the QIR method[27], because most of the conventional global method tend to over-threshold our documents. In [27], a comparison of several global thresholding methods (NIR[27], Otsu [29], Entropy[30] and QIR[27]) finds the QIR method to give the best performance for handwritten images. QIR is a global two step thresholding technique. The first step of the algorithm partition the image into three subimages: the foreground subimage containing pixels darker than A (see Figure 3) , the background part containing pixels brighter than C, and a fuzzy subimage consisting of pixels with gray levels between A and C. For pixels belonging to the fuzzy part it is hard to determine whether a pixel actually belongs to the foreground or the background.

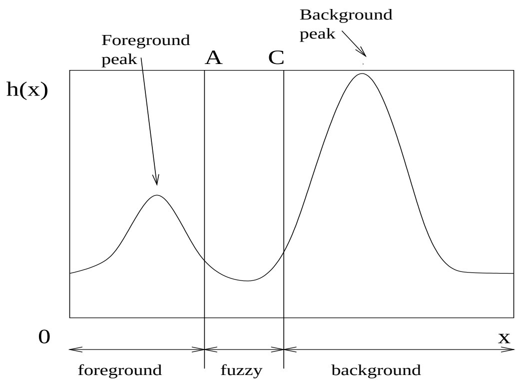


Figure 3. QIR method

The strategy used in the first step is to eliminate all pixels with gray level in the range $[0, A)$ and in the range $(C, 255]$, and thus to consider only pixels in the range $[A, C]$, as possible threshold values. The second step in QIR selects a threshold value according to the writing medium (see [27] for details). For the corrupted documents we work with, characterized by high grayscale variability of foreground objects, sipping ink, and general noise, the QIR method yields suitable under-threshold results for most of our images.

II. Discarding irrelevant objects. After thresholding the document, we omit small blobs and letter punctuation (in Hebrew this are the vowels which are mostly positioned under the text lines). Using a simple line extraction scheme on the binary image provided by the QIR method, the text lines are extracted and the pixels not belonging to lines are deleted. Next we discard all components of the binary image whose areas are smaller than $(mlh*0.15)^2$ where mlh is the average height of the extracted lines. After the cleaning stage, the image components are composed of foreground objects, possibly connected to some noise which will be discarded later. Next we apply *connected component labeling* and split wide components that might consist of merged characters. The written Hebraic text, used in our experiments, is called suqared writing as most of the characters have average width and height approximately equal to the average line height. The overwide components are split to match this criterion. The following step is applied on each of these image components.

III. Local component processing. Since the binarization of the foreground objects is affected only by their local environment, we process information only within the bounding box of each connected component. In the following two steps, we collect the foreground pixels, by first finding a seed set of such pixels and then growing the set according to local neighborhood data.

III.1 Creating the seed image. The first step in this stage is to derive for each component its seed image - all pixels which are classified as foreground pixels. We use a pixel clustering algorithm based on the K-means algorithm[31]. Two clusters are considered, a foreground cluster and a background cluster, according to the gray level. After calculating the foreground and background clusters, we use the average of the gray levels of the foreground cluster as a threshold to generate the seed image consisting of the pixels darker than the threshold..

III.2 The region-growing process. The Growing process is an iterative process, in which during each iteration a set of candidate pixels is observed. Each pixel from this set is tested whether it can join the foreground or not. The process is terminated when no pixel is added to the foreground. The Algorithm goes as follows.

Repeat until the foreground set does not change:

- Find all candidate pixels. The candidate pixels are background pixels which are 8-connected to the growing foreground.
- For each candidate pixel p, consider its 7X7 neighborhood: let M_f be the average grayscale value of the foreground pixels in this window, M_b be the average grayscale value of the background pixels in this window. Assign p to the class whose average is closest to the gray level of p (according to M_f and M_b).

IV. Postprocessing – filling small holes. Due to faded parts of some characters, the component growing process might leave small holes in some characters. In order to avoid filling natural holes which exists in some of the Hebrew characters, all holes with area smaller then $(mlh*0.25)^2$ are filled, where mlh is average line height. Figure 4 shows an input image and the computed binarized image.

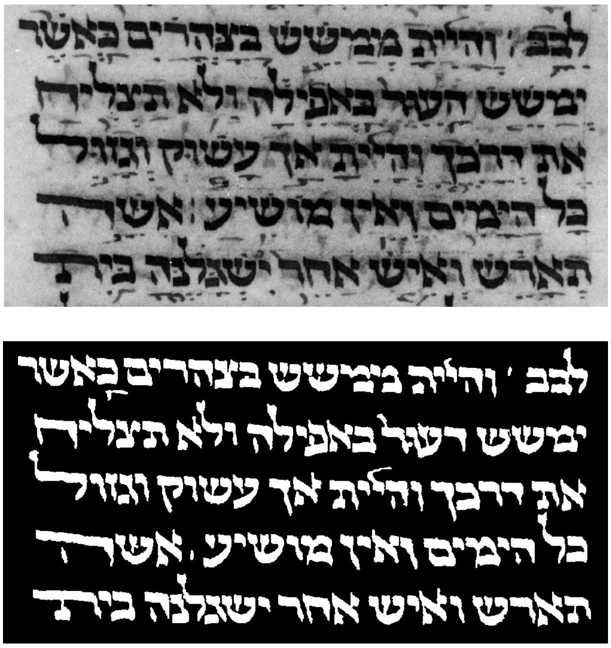


Figure 4. (a) Input image, (b) Binarized image.

3. Feature extraction

The Hebrew alphabet consists of 22 characters. Five of the characters take a different form when appearing at the end of a word.

ש	צ	מ	ל	א
Shin	Tzadi	Mem	Lamed	Aleph

Figure 5. The letters used for style classification.

Character based features are defined for characters that have a more intricate graphic representation, like the characters Aleph, Lmaed, Mem, Tsadi and Shin (see Figure 5). In this paper we concentrate on the letters Aleph and Lamed.

Let B be the binary image of a letter in our document. Denote by S the set of all pixels where $B(i, j) = 1$. The convex hull CHS of the set S is the smallest convex polygon containing S. Let R be the set of pixels belonging to CHS and to the complement of S. $R = \{\text{p} \mid \text{p} \in CHS \ \& \ \text{p} \notin S\}$. Figure 6 shows the sets S, CHS, and R of an Aleph character. As can be seen in Figure 6c, the set R contains a number of disjoint connected components (the white patches in Figure 6c). In each letter, some of the components are of substantial size, and the rest are small ones. We will refer to the large connected components as *dominant background sets*. In the letter Aleph, there are four dominant background sets, while in the letter Lamed, there are two substantial components (see Figure 7). The number of dominant background sets is independent of character size and orientation.

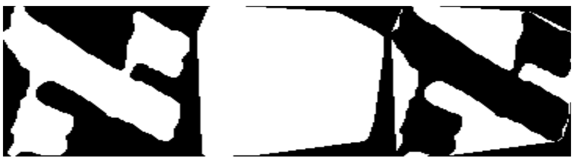
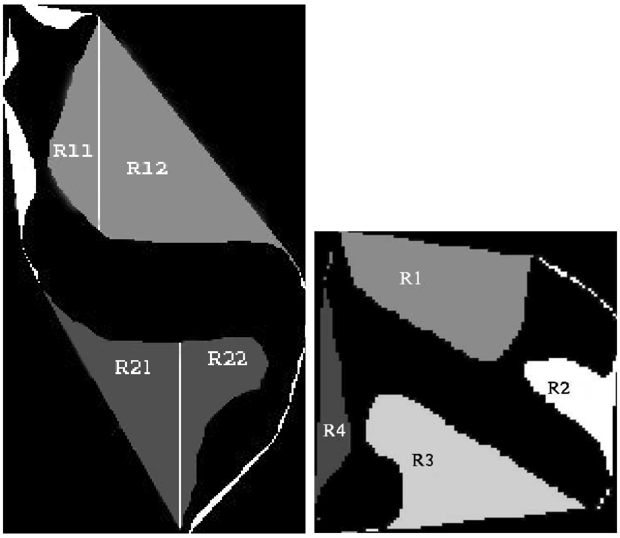


Figure 6. The white pixels are (a) the set S, (b) the set CHS, (c) the set R

Denote the sets of pixels belonging to the dominant background sets by $\{R_i$, i= 1...n$\}$, where n is the number of dominant sets, (four for the letter Aleph and two for the letter Lamed) according to the following:
Let R_1 be the component for which the y coordinate of its center of mass is maximal, and number $R_i, i = 2...n$ in a clockwise order (Figure 7). Denote by $|R_i|$ – the number of pixels in the set R_i.

We use geometric parameters of the sets $\{R_1, \ldots, R_n\}$ as features for classifying the handwrit-



(a) Lamed (b) Aleph

Figure 7. An example for the sets R_i for the letters Aleph and Lamded.

ing style. The following features were used in the preliminary experiment presented here:

Aleph features:

Let $Major(Aleph)$, $Major(R4)$ be the Diameters of the major axis of the ellipse having the same second moment as the character region and R_4 respectivly. Let $Minor(Aleph)$, $Minor(R4)$ be the Diameters of the minor axis of the ellipse having the same second moment as the character region and R_4 respectivly.

- $F_i = \frac{|R_i|}{|CHS|}$, $i = \{1,2,3,4\}$
- $F_5 = \frac{Major(R4)}{Major(Aleph)}$
- $F_6 = \frac{Minor(R4)}{Minor(Aleph)}$

Lamed features:

Let R_{11} and R_{12} be two subsets of R_1, delimited by the vertical line from the highest point of R_1 to the lower base line of the letter. Let R_{21} and R_{22} be two subsets of R_2, similarly delimited by the vertical line starting from the lowest point of R_2 to the upper base line of the letter. Let B_2 be the area of the bounding box of R_2.

- $F_i = \frac{|R_i|}{|CHS|}$ $i = \{1,2\}$
- $F_3 = \frac{|R_{11}|}{|R_1|}$
- $F_4 = \frac{|R_{21}|}{|R_2|}$

- $F_5 = \frac{|R_2|}{|B_2|}$

This is a small subset of the possible variety of features that can be extracted from the sets R of different letters.

4. Experimental Results

Fourteen documents were used in our preliminary experiment. Twenty Aleph and Lamed characters were extracted from each document. At this stage of our experiments, the characters were manually extracted. The experiment was conducted for each letter(Aleph and Lamed) in a "leave fourteen out" manner, as follows: The 280 characters were divided into a training set of 266 characters and a test set of fourteen characters, one from each class. The classification was repeated 20 times, thus each character was classified once. The classification was computed using the Matlab function "Classify", with equal apriory class probabilities, class multivariate Normal density estimation, and linear discriminant functions. The classification results for the letters Aleph and Lamed are summarized in Table 1. As can be seen in Table 1, in each document, the majority of the characters are correctly classified. The aim of our work is document classification, meaning that a document is classified according to the majority of its character classification. In this preliminary experiment 100% correct classification of the documents was achieved.

Doc no.	Correct Alephs	Correct Lameds
d_1	18	11
d_2	13	18
d_3	14	16
d_4	12	13
d_5	11	19
d_6	14	12
d_7	19	15
d_8	17	17
d_9	15	17
d_{10}	13	16
d_{11}	11	14
d_{12}	16	19
d_{13}	20	17
d_{14}	19	18

Table 1. Classification results for documents d_1,...,d_{14}. For each document, 20 characters were classified. As can be seen, for each document the majority of its characters are correctly classifed.

As can be seen in Table 1, in some documents the results for the Aleph are better then the Lameds results, and vice versa. This verifies our hypothesis that different letters better characterize different writers. We plan to develop a classification scheme that will take advantage of this property. A comprehensive study is planned for feature selection and evaluation for Aleph, Lamed and eight more letters. It will then be applied to a much larger database of manuscripts.

5. Conclusion and Future Work

In this paper, we propose methods and present preliminary results for accurate binarization and classification of ancient noisy Hebrew manuscripts. The topological features used for classification in this preliminary experiment are based only on the letters Aleph and Lamed, yielding 100% correct classification of a set of manuscripts written by fourteen writers. We are currently investigating other topological based features to be applied to various letters, and texture based features to be applied to sub-images consisting of a few lines and a few characters in each line, in order to accomplish the following paleographical goals:

1. Dating and localizing the ancient manuscripts.

2. Writer Authentication.

3. Writing-style identification.

References

[1] M. Beit-Arie, Paleographical Identification of Hebrew Manuscripts: Methodology and Practice, in idem, The Making of the Medieval Hebrew Book, The Magnes Press, The Hebrew University, Jerusalem, 1991, pp. 15-44.

[2] G. Nagy, Twenty Years of Document Image Analysis in PAMI, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, January 2000, pp. 38-62.

[3] R. Palmondon and S. N. Srihari, On-Line and Off-Line Handwriting Recognition: A Comprehensive Survey, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, January 2000, pp. 63-84.

[4] K. Etemad, D. Doermann, and R. Chellappa, Multiscale Segmentation of Unstructured Document Pages Using Soft Decision Integration, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 1, January 1997, pp. 92-96.

[5] J. Liang, I.T.Phillips, and R. M. Haralick, An Optimization Methodology for Document Structure Extraction on Latin Character Documents, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 7, July 2001, pp. 719-734.

[6] R. G. Casey and E. Lecolinet, A Survey of Methods and Strategies in Character Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 7, July 1996, pp. 690-706.

[7] L. A. Fletcher and R. Kasturi, A Robast Algorithm for Text String Separation From Mixed Text/Graphics Images, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 10, No. 6, November 1988, pp. 910-918.

[8] G. Nagy and S. Seth, Hierarchical Representation of Optically Scanned Documents, Proceedings of The Seventh International Conference on Pattern Recognition, 1984, pp. 347-349.

[9] G. Agam and I. Dinstein, Adaptive Directional Mathematical Morphology with Applications to Document Analysis. Mathematical Morphology and its Applications to Image and Signal Processing, P. Maragos, R.W. Schafer, and M. A. Butt, Editors, Kluwer Academic Publishers, 1996.

[10] O. D. Trier, A.K. Jain, and T. Taxt, Feature Extraction Methods for Character Recognition - A Survey, Pattern Recognition, Vol. 29, No. 4, 1996, pp. 641-662.

[11] T. Steinherz, E. Rivlin, and N. Interator, Offline Cursive Script Word Recognition - A Survey, International Journal of Document Analysis and Recognition, 1999, pp. 90-110.

[12] L. R. Rabiner, A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, IEEE Proceedings, Vol. 77, No. 2, 1989, pp.257-286.

[13] S. H. Cha and S.N. Srihari,Multiple Feature Integration for Writer Verification, in The Proceedings of the Seventh Workshop on Frontiers in Handwriting Recognition, L. R. B. Schomaker and L.G. Vuurpiji Editors, September 2000, Amsterdam, pp. 333-342.

[14] H.E.S. Said, T.N. Tan and K. D. Baker, Personal Identification Based on Handwriting, Pattern Recognition,vol.33, no.1, pp.149-160, 2000.

[15] Y. Yamazaki and N. Komatsu, A proposal for Text-Indicated Writer Verification Method. Proceedings of ICDAR97

[16] E.N. Zois, and V. Anastassopouls, Fusion of Correlated Decisions for Writer Verification, Pattern Recognition, vol. 32, NO. 10, pp. 1821-1823, 1999

[17] J.M. Fournier and J.C. Vienot, Fourier Transform Holograms Used as Matched Filters in Hebraic Paleography, Israel Journal of Technology, 1971, pp. 281-287.

[18] C. Sirat, L'examen des 'critures: L'oeil et la machine, Paris, Editions du Centre National de la Recherche Scientifique, 1981.

[19] I. Dinstein and Y. Shapira, Ancient Hebraic Handwriting Identification with Run-Length Histograms, IEEE Transactions on Systems Man and Cybernetics, Vol. 12, 1982, pp. 405-409.

[20] L. Likforman-Sulem, H. Maitre, and C. Sirat, "An Expert Vision System for Analysis of Hebrew Characters and Authentication of Manuscripts", Pattern Recognition, Vol. 24, No. 2, 1991,

[21] H.S. Don, A noise attribute thresholding method for document image binarization, Proceeding of 3rd International Conference on Document Analysis and Recognition, Canada, 1995, pp. 231-234.

[22] H. Negishi, J. Kato, H. Hase and T. Watanabe, Character extraction from noisy background for an automatic reference system, Proceedings of 5th International Conference on Document Analysis and Recognition, India, 1999, pp. 143-146.

[23] Y. Liu and S.N. Srihari, Document image binarization based on texture features, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 5, May 1997, pp. 540-544.

[24] C.L. Tan, R. Cao, P. Shen, Q. Wang, J. Chee and J. Chang, Removal of interfering strokes in double-sided document images, IEEE Workshop on the Application of Computer Vision, WACV2000, California, 4-6 Dec 2000, pp. 16-21.

[25] G. Leedham, C. Yan, K. Takru, J. Hadi, N. Tan, L. Mian, Comparison of some thresholding Algorithms for Text/Background Segmentation in Difficult Document Images. Proceeding of 7th International Conference on Document Analysis and Recognition, Scotland, 2003, pp. 859-865.

[26] W. Niblack, An Introduction to Digital Image Proccesing, pp.115-116, Prentice Hall, 1986.

[27] Y. Solihin, C.G.Leedham, Integral Ratio:A New class of Global Thresholding Techniques for Handwritten Images, IEEE Trans. PAMI, vol.21, no. 8, pp.761-768, August 1999. pp. 121-137.

[28] S.D. Yanowitz and A.M.Bruckstein, A new Method for image segmentation, Computer Vision, Graphics and Image Proccesing, vol.46, no.1, pp.82-95, Apr.1989.

[29] N. Otsu A threshold Selection Method from Gray-Level Histogram, IEEE Trans. Systems, Man and Cybernetics, vol 9, pp. 62-66, 1979.

[30] J.N. Kapur, P.K. Sahoo, and A.K.C. Wong, A New Method for Gray-Level Picture Thresholding Using the Entropy of the Histogram, Computer Vision, Graphics and Image Processing, vol .29, pp. 273-285, 1985.

[31] J.B McQueen, Some methods of classification and analysis of multivariate observations, Proc. 5th Berkeley Symposium in Mathematics, Statistics and Probability, vol 1., pp. 281-296, Univ. of California, Berkeley, USA, 1967