Tracing and Straightening the Baseline in Handwritten Persian/Arabic Text-line: A New Approach Based on Painting-technique

P. Nagabhushan and Alireza Alaei
[1,2]Department of Studies in Computer Science, University of Mysore, Mysore, 570 006, India
[1]alireza20alaei@yahoo.com,[2]pnagabhushan@hotmail.com

***Abstract*—In this research work, we propose to identify an imaginary line called baseline threading through the entire stretch of text-line, with reference to which the location of vertical extents of Persian characters could be accurately interpreted. Depending upon the curvedness of the handwritten Persian text-line the baseline also would be curved. In this research a novel piece-wise painting scheme is proposed to prepare patches of black and white blocks all along the text-line, identify some candidate points, regress a curve through these candidate points to trace the baseline which is subsequently stretched straight horizontally and subsequently we de-tilt the characters to align the text-line with the horizontal imaginary baseline properly. The proposed algorithm is evaluated with 108 Persian handwritten text-lines containing 3612 subwords. Experimental analysis showed that 91.2% of the subwords are accurately aligned. Further, the proposed scheme is tested with another dataset containing 600 text-lines [13] and more accurate results are achieved when compared with the results reported in state of the art for the same dataset. The effectiveness of aligning text-lines linearly is demonstrated through OCRing for readability of tilted printed English text-lines and corresponding transformed text-lines, which are obtained using the proposed procedure.**

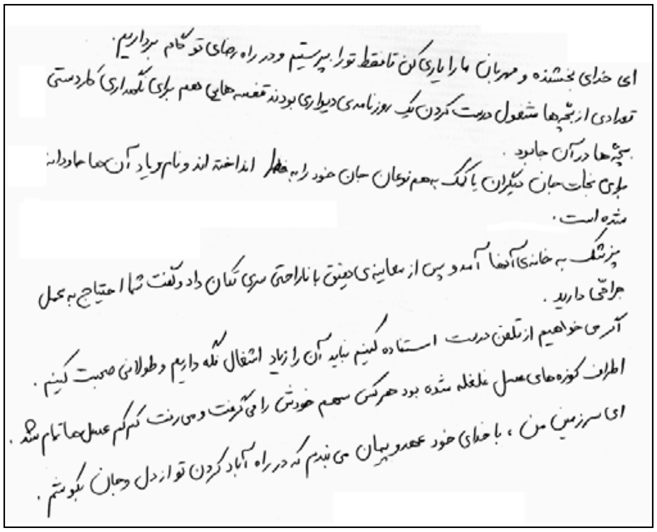
Keywords: Handwritten Document; Baseline; Piece-wise Painting; Curve Fitting.

I. INTRODUCTION

In any OCR system, preprocessing is the primary step that plays an important role in reliable segmentation, feature extraction, and finally recognition. Skew detection/ correction as a preprocessing technique, is widely studied/employed for enhancement of different OCR results [1]. However, in an unconstrained handwritten text-page written in any language especially Persian and all similar scripts every text-line may not be only skewed, it also would suffer a slow varying oscillation (non-linear alignment). Moreover, a skew detection/correction technique, which results in a global orientation for entire text-page/text-line, cannot be utilized in case of an oscillatory text-line as a preprocessing operation. This becomes a bottleneck in machine processing of handwritten text documents. The problem would be more acute in case of Persian text-lines because of natural cursiveness associated with the characters and the vertical displacement permitted in placing character segments. A Persian handwritten text-page with different skews in different text-lines is shown in Fig.1, which clearly shows the limitation of global skew detection/correction. Moreover, in any OCR system subsequent to text-line segmentation, word/character level segmentation is a mandatory process to achieve the goal. Word/character level segmentation of such oscillatory text-line is very complex and crucial task [2], [3]. A wide study on text-line segmentation as well as recent approaches for segmenting a text-page into corresponding text-lines is presented in [4], [5]. However, in case of Persian handwritten recognition most of the research works have not gone beyond text-line segmentation. To facilitate segmentation of a Persian text-line up to character level, it is necessary to utilize some specific characteristics of Persian script in different stages of Persian document analysis and recognition [8], [9]. We illustrate these particular characteristics of Persian scripts in this section to get an idea about them, which can specially be used in preprocessing phase.

Persian (Farsi) has 32 basic characters and each character can have up to four forms. Each Persian, Arabic, or Urdu word/subword consists of two or more characters. In a ruled paper Persian text-lines, which contain a group of words/subwords/characters, are placed with reference to the apriori available printed lines. In an un-ruled paper, a Persian text-line is however assumed to have been placed with reference to an imaginary line called baseline. This characteristic is one of the main properties of text-lines in Persian and other similar scripts. The concept/importance of baseline/reference line is also presented in pitman shorthand text analysis and recognition since strokes are typically written on ruled papers. However, locating the reference lines when Pitman Shorthand Language (PSL) is written on a white paper is a crucial problem in PSL [6], [7]. To demonstrate the concept of baseline, horizontal projection profiles for perfectly horizontally aligned printed text-lines in different languages are calculated. By looking at the results of horizontal projection profiles presented in Fig.2-(a..d), it is clearly revealed that there is no relevance between the respective base lines in the printed English, Hindi and Kannada text-lines, and positions of peak values in their projection profiles. However, there is only one smooth and long peak in the horizontal projection profile of Persian text-line, which aligns with the position of baseline. This characteristic of correspondence/ matching between position of baseline and the peak value in the projection profile can be used in Persian document analysis and recognition [8], [9].

In most of the research papers related to Persian/Arabic character segmentation [8], [9], researchers have used several techniques to detect the baseline of the word/subword or line in the preprocessing step to use this information in subsequent steps. It is shown that the baseline information improved both the segmentation and recognition accuracies [8], [9].



ای خدای بخشنده و مهربان ما را یاری کن تا فقط تو را بپرستیم و در راه رضای تو گام برداریم.
تعدادی از بچه‌ها مشغول درست کردن یک روزنامه‌ی دیواری بودند قفسه هایی هم برای نگهداری کاردستی
بچه‌ها در آن جا بود.
برای نجات جان دیگران با کمک به هم نوعان جان خود را به خطر انداخته اند و نام و یاد آن‌ها جاودانه
شده است.
پزشک به خانه‌ی آن‌ها آمد و پس از معاینه‌ی دقیق با ناراحتی سری تکان داد و گفت شما احتیاج به عمل
جراحی دارید.
اگر می خواهیم از تلفن درست استفاده کنیم نباید آن را زیاد اشغال نگه داریم و طولانی صحبت کنیم.
اطراف کوزه‌های عسل غلغله شده بود هر کس سهم خودش را می گرفت و می رفت کم کم عسل‌ها تمام شد.
ای سرزمین من، با خدای خود عهد و پیمان می بندم که در راه آباد کردن تو از دل و جان بکوشم.

Fig.1. A Persian handwritten text-page with different skews in different text-lines

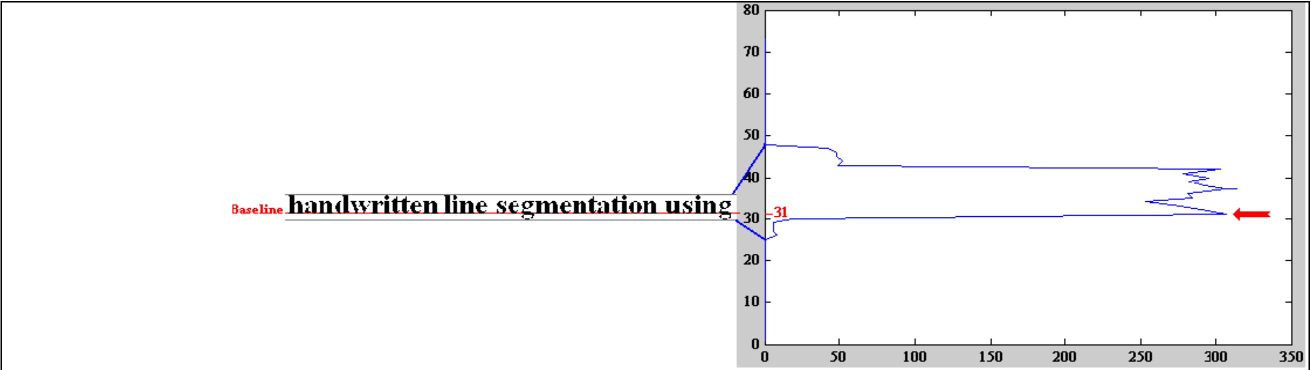


Fig.2-a. A printed English text sample and its horizontal projection profile (Peak value ≠ Baseline)

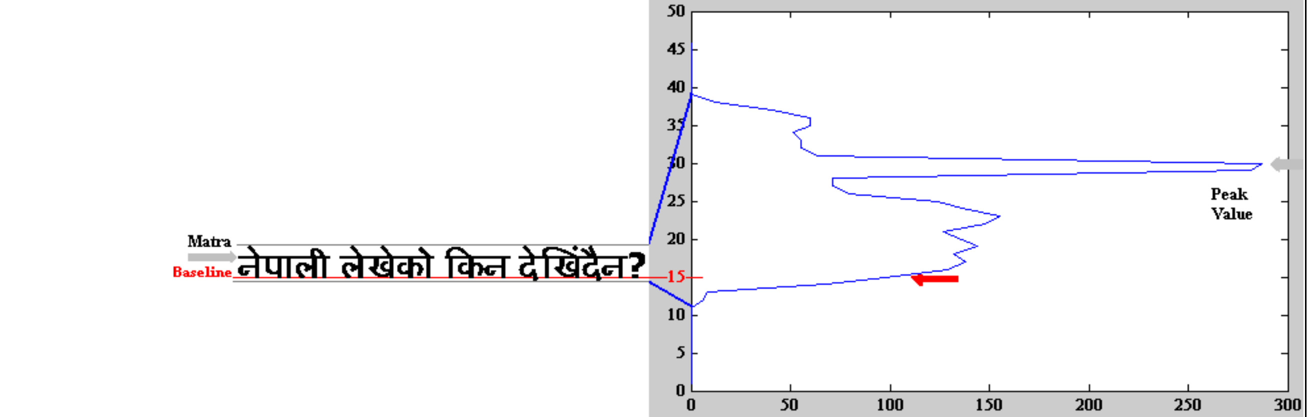


Fig.2-b. A printed Hindi text sample and its horizontal projection profile (Peak value = Position of Matra or Peak value ≠ Baseline)

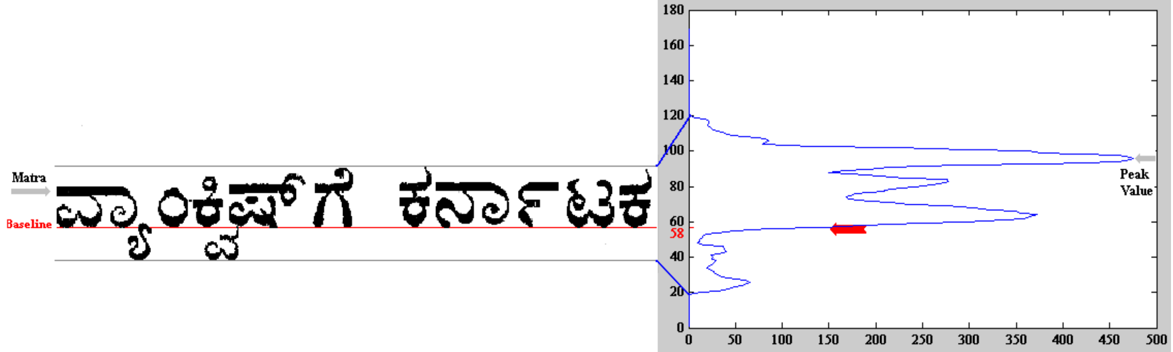


Fig.2-c. a printed Kannada text sample and its horizontal projection profile(Peak value = Position of Matra)

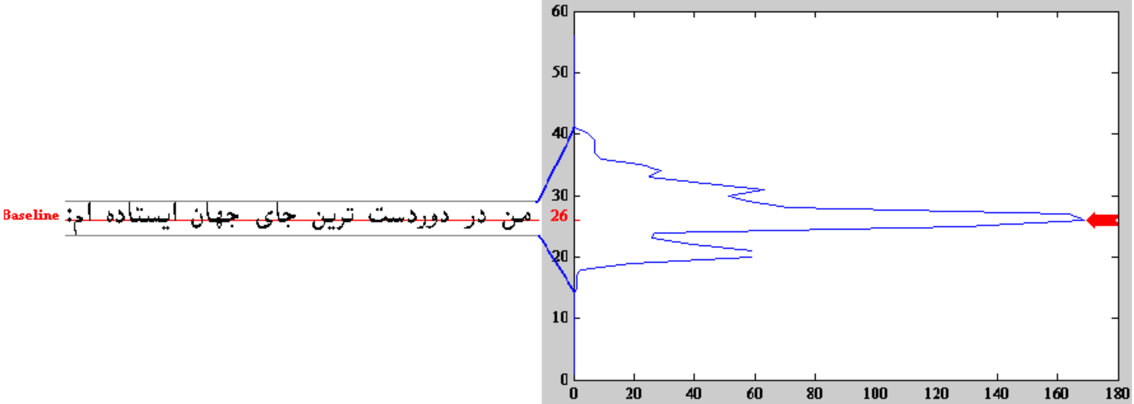


Fig.2-d. a printed Persian text sample and its horizontal projection profile (Peak value = Baseline position)

Note: Lines with Black color indicate baselines (reference line) in the text-lines printed with different languages and Black color arrows indicate position of baseline in horizontal projection profiles

In [8], [10], baseline is detected by finding peak value of horizontal projection profile in a printed/handwritten text-line. In [3], a modified projection technique is addressed. In this technique by rotating word image through different angular inclinations, the peak value of horizontal projection profile in each angular position is obtained and the maximum value and corresponding angle among all the peak values is found. Subsequently by rotating the word with the obtained angle, the baseline of Arabic word is identified. In [11], the concept of baseline is used to find the best positions for Arabic character segmentation. In [9], parameters such as lower/upper baselines and distance from the uppermost text pixel to the baseline and distance from the lowermost text pixel to the baseline are used to derive a subset of geometrical baseline dependent features. It is shown that accurate baseline has direct effect in the feature extraction and recognition methodology [9]. In [12], the baseline is used for rough line separation and then by using contour information the exact lines have been extracted. Recently a two-stage Persian/Arabic baseline detection and correction algorithm is presented in [13]. The first stage estimates the writing path of a text-line by a fitted curve based on candidate baseline pixels, which are detected using template matching algorithm. Then the slant and position of the components in the line is adjusted. In the second stage, the baseline for each subword is corrected.

However, the horizontal projection profiles [8], [10] and its modified version [3] cannot be directly utilized for tracing (detecting) the baseline in Persian/Arabic handwritten text-line because each text-line generally suffers with oscillatory placements of words/ subwords with respect to the imaginary baseline. The template matching technique presented in [13] results in improper candidate points in many cases, which consequently provides inappropriate baseline alignment. Moreover, because of curssiveness of handwritten Persian texts, it is very difficult to see text-lines maintaining perfect horizontally straight path. The upwards/downwards direction skewed text-line could be the simplest setback; however more severe would be when a text-line suffers oscillatory placement, which the direction of text-line frequently changing both upwards and downwards. These difficulties and presence of some special properties in handwritten Persian text such as various shapes for a group of two or three dots like horizontal bar and circle, which appear above/below the main body of the character far from the baseline [3] make tracing/aligning the baseline more complex and a challenging problem. In addition, the technique presented in [13] as authors themselves have mentioned, is a time-consuming preprocessing operation. In order to overcome such difficulties we propose a novel framework for tracing and straightening the baseline in Persian handwritten text-line, which can also be used for other similar scripts in order to facilitate word/character segmentation. In the present work, a new painting technique is proposed to prepare patches of black and white blocks all along the text-line and subsequently identify candidate points for tracing the approximate baseline. Curve fitting is utilized to approximately trace the baseline by fitting a curve through the extracted candidate points. By using the candidate points (pixels) presented in each component, corresponding average slope for counter rotating the component is calculated. Each component presented in the text-line is rotated by respective average slope to approximately align with the imaginary baseline. Further, to achieve more accurate baseline, each component is lined up with respect to imaginary baseline.

The rest of the paper is organized as follows: A new model for baseline tracing is described in Section 2. Section 3 presents the baseline alignment methodology. Experimental results and discussion are reported in Section 4. Finally, conclusions and future work are presented in Section 5.

II. A New Model for Baseline Tracing

To trace the baseline, it is necessary to find areas, which are very close to the baseline and these areas are called key areas. Therefore, the input text-line (gray or BW image) is converted into an image of painted strips (Fig. 3 & 4) using the algorithm presented in [4], [14]. This algorithm [4], [14] works based on decomposing the image into a number of vertical strips with a certain width from the right side. Then every pixel value in each row of the strip is replaced by a gray intensity, which is the average intensity value of gray values of all pixels present in that row of the strip. Subsequently, the scripts are converted into two-tone painting. The result of this step, which is an image with a number of rectangular black blocks, is shown in Fig.4. In the painted image, the black blocks with maximum height are chosen in every vertical strip (Fig. 5). These blocks, which are called key blocks (areas), help finding points (pixels), which are very close to the baseline. Then the dot(s) and strokes are removed from the input image based on sizes of components present in the input text-line (Fig. 6). The reasons behind this operation are: the dot(s) or strokes are usually positioned in far locations from main body of words/subwords or characters and they make the process of baseline tracing more complicated. Then by mapping the input text-line image without dots and strokes on the black blocks with maximum heights (key blocks) the key portions in the text-line are found. Piece-wise horizontal projection in the entire region of each key black block is calculated and maximum piece-wise horizontal projection is found respectively. Based on the maximum piece-wise horizontal projections, the points (pixels) which belong to text portion in each block are chosen as candidate points.

In unconstrained handwritten text document, each text-line has different flow of writing and even it may not be a straight skewed line. Therefore, in the present work, it is decided to fit a curve on input text-line by utilizing polynomial fitting algorithm [15]. Formula for polynomial with degree of n is shown in (1).

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_2 x^2 + a_1 x + a_0 \quad (1)$$

Where n is a nonnegative integer and a0, a1, a2, ..., an are constant coefficients and f(x) is defined for all values of x. In the present work the (x, y) coordinates of candidate points are mapped with (x, f(x)) in (1) respectively. The degree of polynomial (n) is considered as 4 based on an assumption that the oscillation of a handwritten text-line will not be more than 3. So in case of oscillation more than 3 in a text-line a curve is fitted with degree of 4 on the input text-line. Moreover, we tested the system with 108 text-lines of Persian handwritten using the degree 3, 4, and 5 for the degree of polynomial and we found better experimental result with degree of 4. Subsequently the polynomial coefficients are also calculated using extracted candidate points. The result of curve fitting operation is shown in Fig.7-(a) and corresponding coefficients are tabulated in Table 1. This curve is further used to align the oscillatory written input text-line into a straight horizontal text-line, which facilitates subsequent processes. More results of curve fitting operation are shown in Fig.7-(b..f) and corresponding coefficients of each curve are also tabulated in Table 1. The highest value among the values of column a4 in Table 1 belongs to text sample 1. It is evident that the text sample 1 is more complex when compared with the other text-lines. By looking at Table 1, it is also clear that for some text-lines especially text sample 3, a curve of degree 3 is good enough to be fitted on its candidate points.

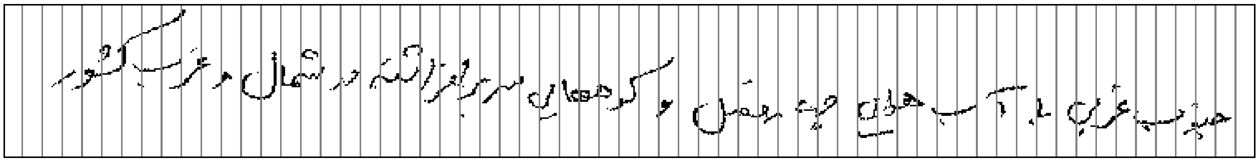


Fig.3. A Persian text-line image after decomposing it into vertical strips

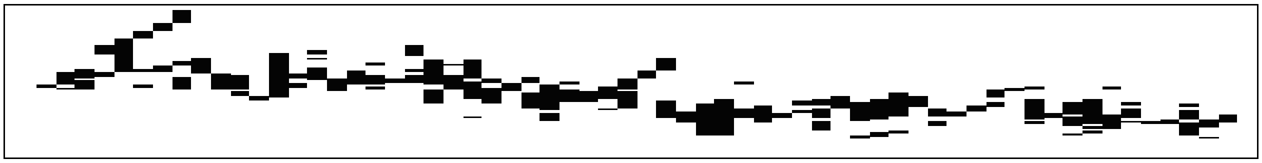


Fig.4. Painted image after applying painting algorithm

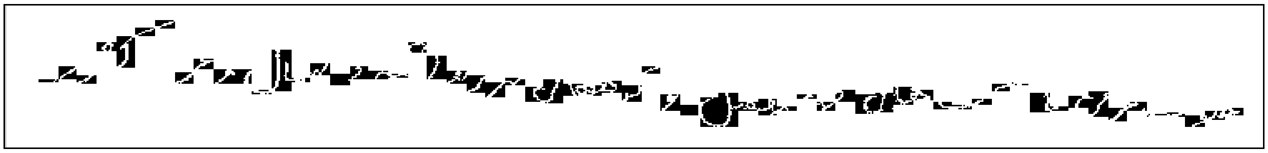


Fig.5. Text-line image after selecting black rectangles with maximum heights in each strip and mapping the original text-line image on the selected black areas and finding candidate points

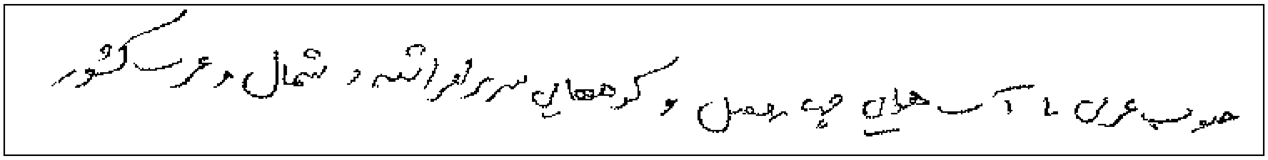


Fig.6. Text-line image after removing dots and strokes

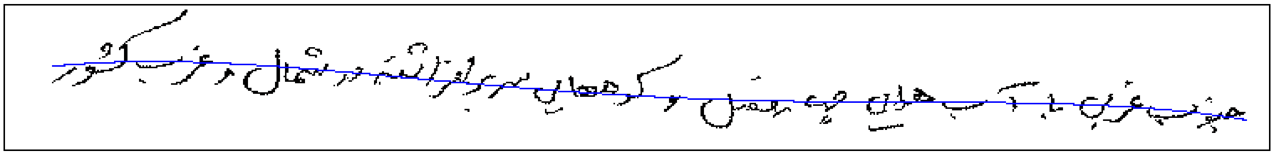


Fig.7-a. Fitted curve in the original image (Text-line sample 1)

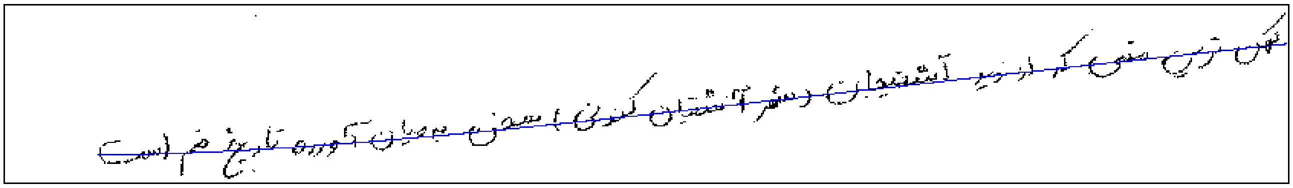


Fig.7-b. Fitted curve in the handwritten text-line sample 2

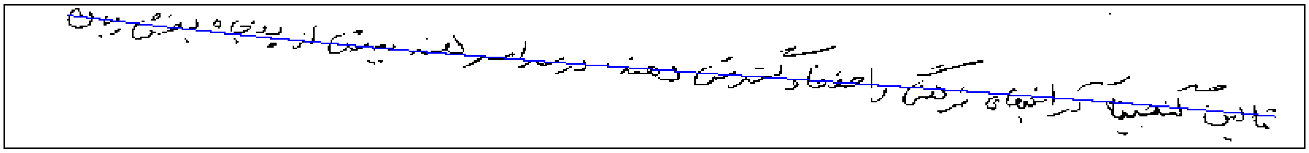


Fig.7-c. Fitted curve in the handwritten text-line sample 3

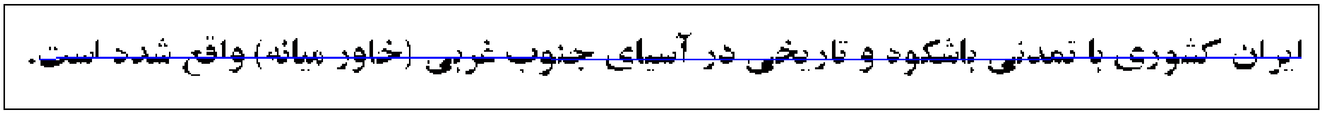


Fig.7-d. Fitted curve in the printed text-line sample 4 (without skew)

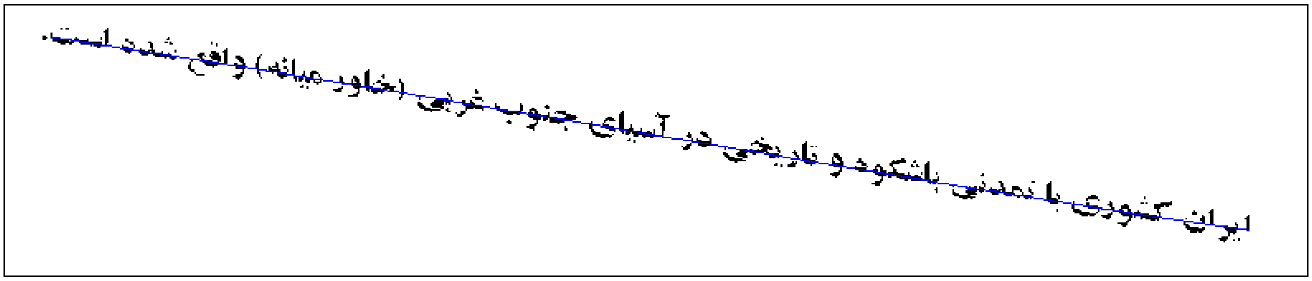


Fig.7-e. Fitted curve in the printed text-line sample 5 (Skewed upwards)

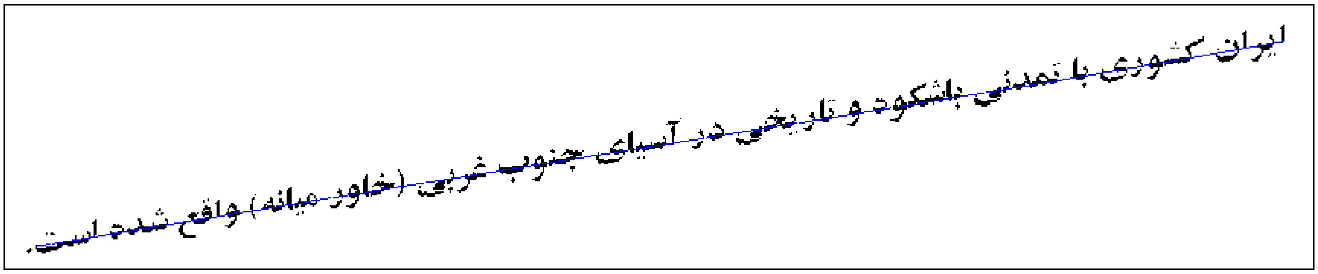


Fig.7-f. Fitted curve in the printed text-line sample 6 (Skewed downwards)

TABLE. I. POLYNOMIAL COEFFICIENTS OF DIFFERENT TEXT-LINES

	a_4	a_3	a_2	a_1	a_0
Sample 1	0.00000000007	-0.00000031583	0.00047564526	-0.19046234819	135.39963824961
Sample 2	-0.00000000001	0.00000005569	-0.00014227586	0.03197629125	289.12796011878
Sample 3	0.00000000000	0.00000000650	-0.00003880138	0.13563393873	6.04533525788
Sample 4	0.00000000004	-0.0000009987	0.00007183965	-0.01690945459	57.10543179244
Sample 5	-0.00000000002	-0.00000000698	0.00003790800	0.16030916415	34.90633719825
Sample 6	-0.00000000002	0.00000000171	0.00001686790	-0.18181654990	172.21903233955

III. BASELINE ALIGNMENT METHODOLOGY

An unconstrained handwritten text-line suffers from a slow varying oscillation in its components. To find the slope for each component in the text-line instead of finding a global slope for entire text-line, we use some language based knowledge such as direction of writing in Persian and similar scripts, which is from right to left. As a consequence of this influence, subsequent to fitting a curve on the input text-line, the rightmost point of the fitted curve is found. This point is indicated with an up arrow in Fig.8. Further, all intersection points, which occur between the curve drawn and the components (words, subwords, dots and strokes) presented in the input text-line, are obtained. The intersection points in each component are used to find slope of component with respect to horizontal line. To obtain the average slope for each component, a hypothetical horizontal line passing from the rightmost point of the fitted curve is drawn (Fig. 9 and 10). From the rightmost point, an additional straight line is drawn to each intersection point in the corresponding component. Slope between these lines and hypothetical horizontal line are calculated respectively and average of them is obtained. In reality, different pixels in a component should be rotated with different slopes; calculation of slopes for all pixels in a component is a tedious task. Therefore, for every component the average slope value is calculated separately. Each component is rotated with its own average slope to align with reference to the imaginary baseline. As it is shown in Fig.9 and 10, if we assume there are two intersection points in the component indicated by circle, it is needed to draw two straight lines between these two points and the rightmost point. The average value of the slope β1 and β2, which is [(β1+β2)/2], is the corresponding slope for rotating the component indicated by circle. In addition, there are a number of components (shown with rectangles in Fig. 8), which do not have any intersection with the fitted curve in the text-line. To rotate these components with accurate slope a distance matrix is constructed based on the distances from the center of gravity of every component to the center of gravities of all other components. The corresponding slope of such component is obtained by finding the minimum distance in corresponding row of the distance matrix. Figure 11 shows the result of the first step of baseline alignment.

Figures 14 and 15 clearly show the significance of the proposed technique on shapes of horizontal projections in original input text-line image (Fig. 3) and the image after the first stage of baseline alignment (Fig. 11).

As it is demonstrated in Fig.12, some components are not placed in proper positions with respect to proposed baseline. Therefore, it is necessary to shift those components in the appropriate position to get more accurate baseline. To do so for every component in the input text-line, horizontal projection is computed and maximum horizontal projection value and its position are found. Then the components, which have at least one intersection point with the drawn baseline and place very close to the baseline, are vertically shifted upward or downward. The difference between the position of the drawn horizontal baseline and position of baseline of each component is measured as value of shifting. The baseline of the each component may be placed in the position of maximum horizontal projection or end-points (uppermost/ lowermost).

For the components, which do not have any intersection point with the drawn baseline, distance matrix is calculated in the same manner as done earlier. For each such component minimum distance with respect to the components having intersection point(s) with the drawn baseline is obtained and respective vertical movement is considered for vertically shifting up/down the respective component.

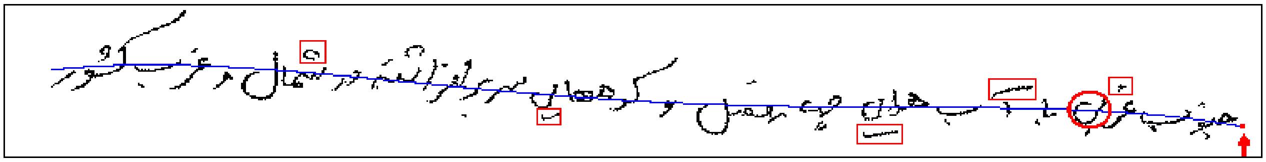


Fig. 8. Rightmost point (indicated by arrow), a component having intersection points with fitted curve (indicated by circle) and some components that do not have any intersection with fitted curve (indicated by rectangles)

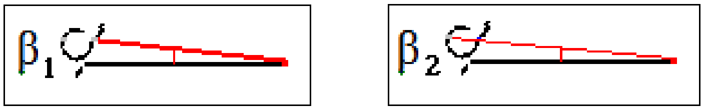


Fig. 9 and Fig. 10. Slopes β1 and β2 between horizontal line and lines drawn from rightmost point to two different intersection points in corresponding component.

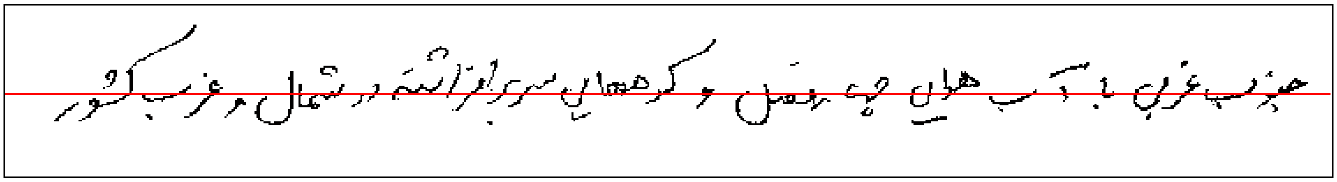


Fig. 11. Result of baseline aligning technique after aligning the component in first step

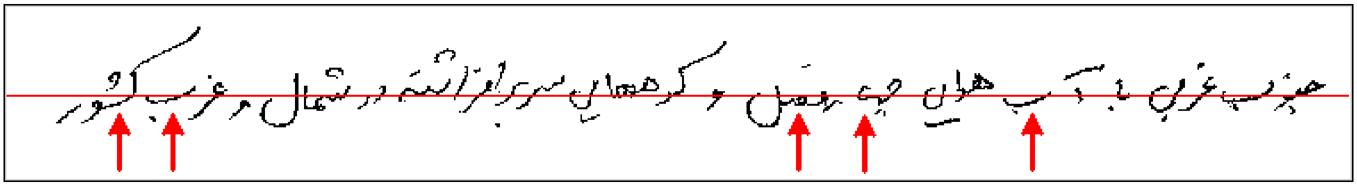


Fig.12. Components, which need more alignment with respect to baseline

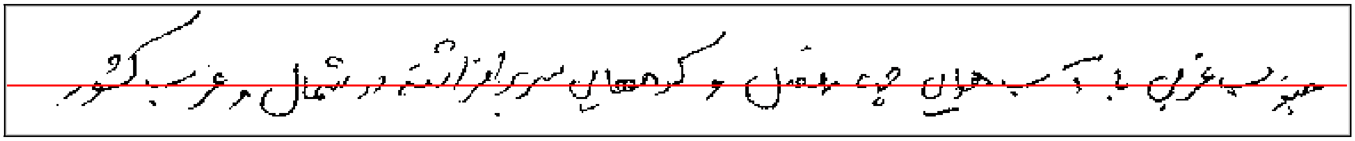


Fig.13. Result of the proposed algorithm

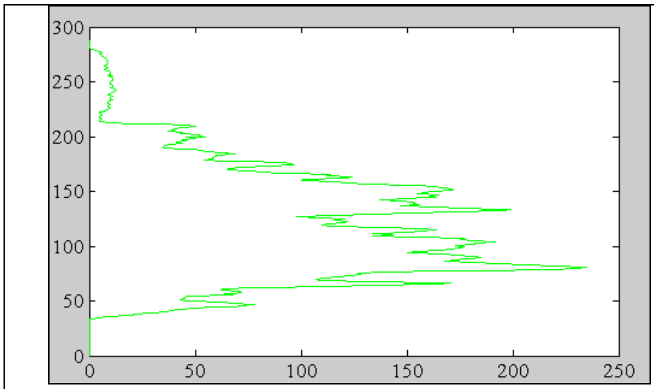


Fig. 14. Horizontal projection of original image

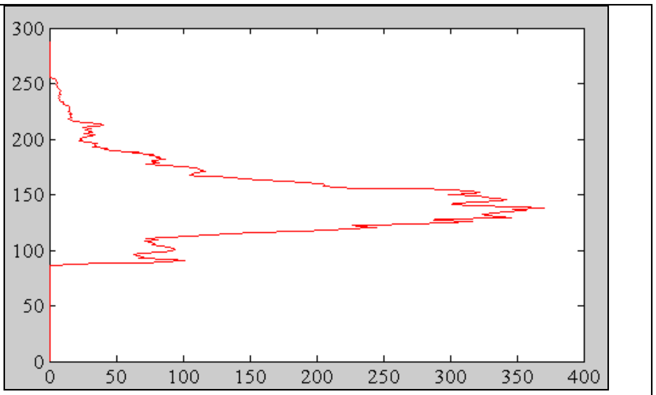


Fig. 15. Horizontal projection of the result after aligning the component in first step

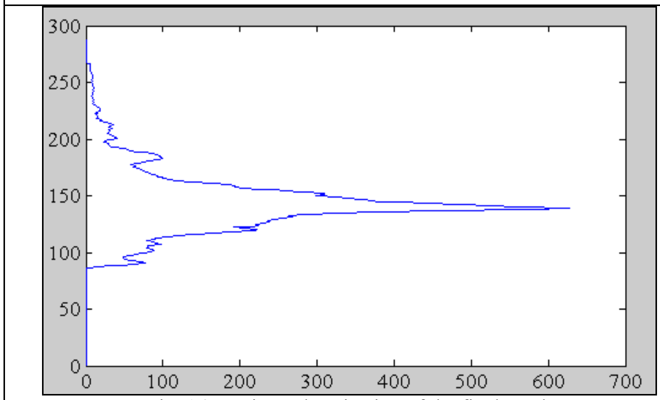


Fig. 16. Horizontal projection of the final result

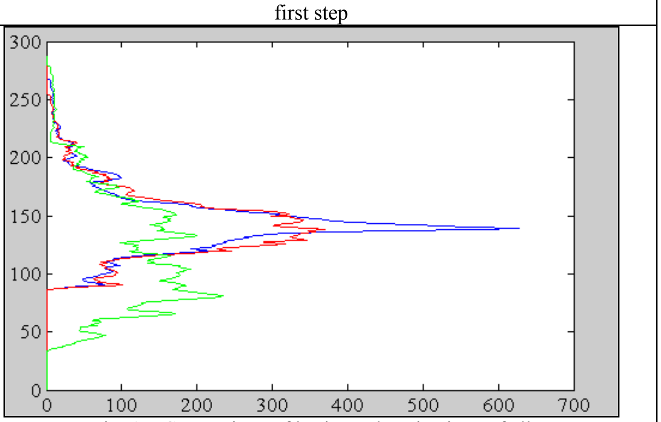


Fig. 17. Comparison of horizontal projections of all The projection profiles

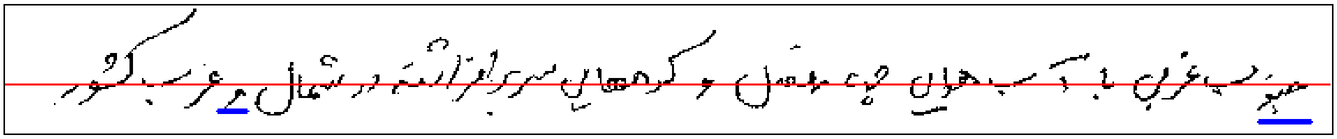


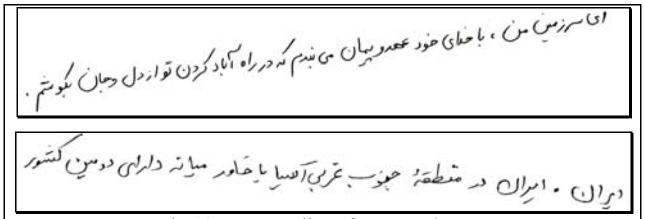
Fig. 18. Misaligning character/characters in a text-line after baseline alignment

IV. Experimental Results and Comparison Analysis

The proposed algorithm is tested with 108 handwritten Persian text-lines containing 3612 subwords written by different writers and scanned in gray level with resolution of 300 DPI. For each text-line, accuracy of the proposed algorithm is computed as the ratio of number of subwords correctly aligned with respect to the baseline to total number of subwords in the text-line. (e.g. in Fig. 8 the accuracy of the proposed algorithm for aligning the baseline is 30/32=93.75% that means 30 subwords out of 32 subwords present in the text-line are correctly aligned). We manually evaluated the accuracy in each text-line. Finally, average value of the accuracies in all the text-lines is considered as overall accuracy of the proposed algorithm. From experimental results, 91.2% of the subwords are correctly aligned with respect to the actual baseline. Figures 14-16 show the results of horizontal projection in original image, image after first level of aligning the baseline and image after second level of baseline alignment. Considerable improvements in aligning the baseline can be observed in figures 14-17. The underlined subwords in Fig. 18 are the subwords misaligned with respect to actual baseline in the chosen sample text-line. To the best of our knowledge, there is no handwritten Persian text-line dataset with ground truth information to help us for automatic evaluating the present work and comparing our results with literature results. Therefore, we tested the proposed scheme with the dataset introduced in [13]. The dataset consists of 600 text-lines containing 14,600 subwords [13] without ground truth information. We fixed margin parameter, which was defined as maximum allowable distance of aligned baseline from the actual baseline of subwords [13], to 15 for having a proper comparison. In subword level, we manually evaluated the present work and out of 14600 subwords 14058 subwords (96.29% of the subwords) were correctly aligned with respect to their baselines. From the work presented in [13] the result of baseline correction with the same dataset and the same margin parameter was reported to be 95.12%. From the experimental results, it is clear that the proposed algorithm shows better results when compared with the work presented in [13].

The present work showed lower accuracy (91.2%) with our dataset. This is because our text-line dataset is more complex then the dataset introduced in [13]. To have the idea about the types of text-lines in these two datasets some text-lines are shown in Fig.19 and Fig.20.

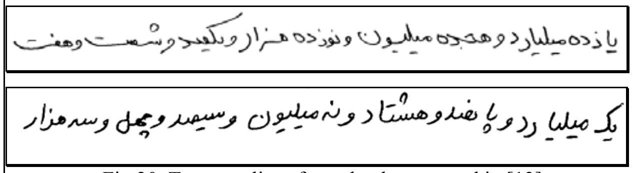
To show applicability of the proposed framework on other languages a few lines of printed English text with different contents and skews from 10 to 20 degrees are considered. We employed a tool (Readiris Pro 10) for converting these text images into corresponding word file format. We manually evaluated the results by counting the number of characters, which are correctly recognized by the OCR tools. By applying only the first stage of proposed framework and then employing OCRing process, accuracy has significantly improved from 40.03% to 98.01% in the final OCRs' result. Some results of the proposed technique are shown in Fig. 21.



ای سرزمین من ، با خاک خود عهد و پیمان می بندم که در راه آباد کردن تو از دل و جان بکوشم .

ایران . ایران در منطقهٔ جنوب غربی آسیا با خاور میانه دارای دومین کشور

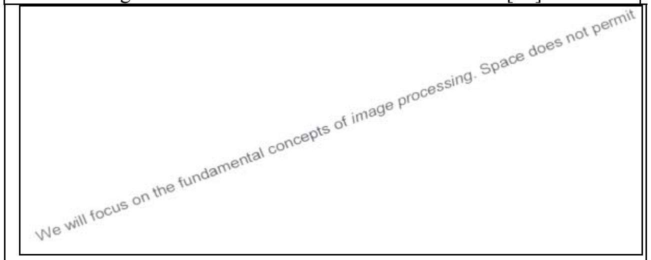
Fig.19. two text-lines from our dataset



یازده میلیارد و هجده میلیون و نوزده هزار و یکصد و شصت و هفت

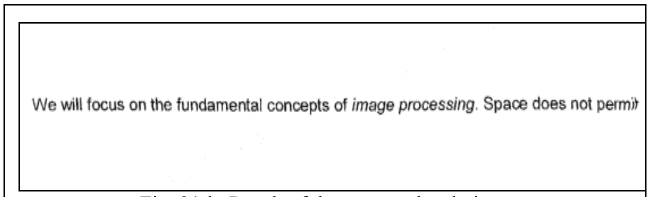
یک میلیارد و پانصد و هشتاد و نه میلیون و سیصد و چهل و سه هزار

Fig.20. Two text-lines from the dataset used in [13]



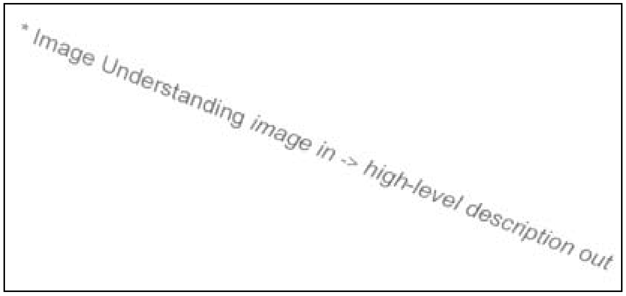
We will focus on the fundamental concepts of *image processing*. Space does not permit

Fig. 21-a. Original printed English text-line



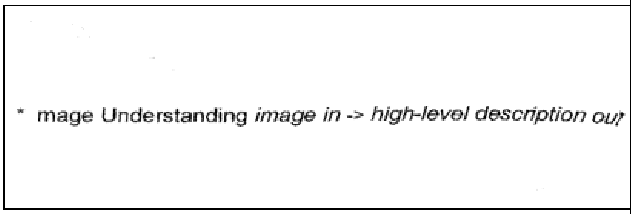
We will focus on the fundamental concepts of *image processing*. Space does not permi

Fig. 21-b. Result of the proposed technique



* Image Understanding *image in -> high-level description out*

Fig. 21-c. Original printed English text-line



* mage Understanding *image in -> high-level description ou*

Fig. 21-d. Result of the proposed technique

V. CONCLUSION

In this paper, a novel algorithm for baseline tracing and alignment in Persian handwritten text-line is introduced. The proposed algorithm is evaluated by 108 Persian text-lines and 91.2% of the subwords in our test dataset (text-lines) are placed in proper location with respect to the actual baseline. The present work is shown better result when compared with the results of the literature. The main advantage of the present work is very quickly finding the candidate points for baseline tracing in the input text-line image and finding the slope for each component in the text-line and straighten it instead of finding a global slope for entire text-line. Moreover, it is aimed that the proposed algorithm as a preprocessing technique can facilitate segmentation of a text-line into words/subwords and characters with more ease and accuracy.

In future, we plan to work on automatically determination of the degree of polynomial for best curve fitting and extend the work by using piece-wise alignment, which may help us to achieve better results.

REFERENCES

[1] J. J. Hull, "Document image skew detection: Survey and annotated bibliography," Document Analysis Systems II, J. J. Hull and S. L. Taylor (Eds), World Scientific, 1998, pp. 40-64.

[2] A. AL-Shatnawi and K. Omar, “Methods of Arabic Language Baseline Detection – The State of Art,” International Journal of Computer Science and Network Security, vol.8, no.10, Oct. 2008.

[3] Hasan Al-Rashaideh, “Preprocessing phase for Arabic Word Handwritten Recognition,” Electronic Scientific Journal, vol. 6, no. 1, 2006, pp. 11-19.

[4] B. Gatos, N. Stamatopoulos and G. Louloudis, “ICDAR2009 Handwriting Segmentation Contest,” Proc. of 10th International Conference on Document Analysis and Recognition (ICDAR'09), 2009, pp.1393 – 1397.

[5] B. Gatos, A. Antonacopoulos and N. Stamatopoulos, "ICDAR2007 Handwriting Segmentation Contest,” Proc. of 9th International Conference on Document Analysis and Recognition (ICDAR'07), 2007, pp.1284-1288.

[6] P. Nagabhushan and S. Murali, “Recognition of Pitman shorthand text using tangent feature values at word level,” Sadhana, vol. 28, no. 6, Dec. 2003, pp. 1037-1046.

[7] P. Nagabhushan and B. S. Anami, “A knowledge-based approach for recognition of handwritten Pitman shorthand language strokes,” Sadhana, vol. 27, no. 5, Dec. 2002, pp. 685-698.

[8] R. Azmi and E. Kabir, “A new segmentation technique for omnifont Farsi text,” Pattern Recognition Letters 22, 2001, pp. 97-104.

[9] R. El-Hajj, L. Likforman-Sulem, and C. Mokbel, “Arabic Handwriting Recognition Using Baseline Dependant Features and Hidden Markov Modeling,” Proc. of Eighth International Conference on Document Analysis and Recognition (ICDAR05), 2005, pp. 893-897.

[10] A. M. Elgammal and M. A. Ismail, “A Graph-Based Segmentation and Feature Extraction Framework for Arabic Text Recognition,” Proc. of the Sixth International Conference on Document Analysis and Recognition (ICDAR01), 2001, pp. 622-626.

[11] D. Motawa, A. Amin, R. Sabourin, “Segmentation of Arabic Cursive Script,” Proc. of the Fourth International Conference on Document Analysis and Recognition (ICDAR97), 1997, pp. 625-628.

[12] B. Yanikoglu and P. A. Sandont, “Segmentation of Off-Line Cursive Handwriting Using Linear Programming,” Pattern Recognition, vol. 31, no. 12, 1998, pp. 1825-1833.

[13] M. Ziaratban and K. Faez, “A Novel Two-Stage Algorithm for Baseline Estimation and Correction in Farsi and Arabic Handwritten Text line,” Proc. of International Conference on Pattern Recognition (ICPR’08) , 2008, pp. 1-5.

[14] P. Nagabhushan and A. Alaei, “Unconstrained Handwritten Text-line Segmentation Using Morphological Operation and Thinning Algorithm,” Proc. of IICAI09, 2009, pp. 2080-2091.

[15] Coope, I.D., “Circle fitting by linear and nonlinear least squares,” Journal of Optimization Theory and Applications, vol. 76, no. 2, Feb. 1993, pp. 381-388.