2018 24th International Conference on Pattern Recognition (ICPR)
Beijing, China, August 20-24, 2018

Weighted-Gradient Features for Handwritten Line Segmentation

[1]Vijeta Khare, [1]Palaiahnakote Shivakumara, [2]B. J, Navya, [2]G. C, Swetha, [2]D. S. Guru, [3]Umapada Pal and [4]Tong Lu

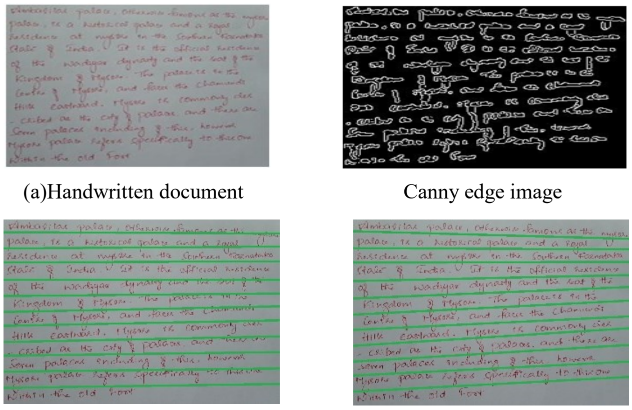
[1]Faculty of Computer Science and Information Technology, University of Malaya, Kuala Lumpur, Malaysia.
Email: kharevijeta@gmail.com, shiva@um.edu.my.
[2]Department of Studies in Computer Science, University of Mysore, Karnataka, India.
Email:navyagowdab.j@gmail.com,swethagc46@gmail.com, dsguruji@yahoo.com
[5]Computer Vision and Pattern Recognition Unit, Indian Statistical Institute, Kolkata, India. Email: umapada@isical.ac.in
[4]National Key Lab for Novel Software Technology, Nanjing University, Nanjing, China. Email: lutong@nju.edu.cn.

***Abstract*—Text line segmentation from handwritten documents is challenging when a document image contains severe touching. In this paper, we propose a new idea based on Weighted-Gradient Features (WGF) for segmenting text lines. The proposed method finds the number of zero crossing points for every row of Canny edge image of the input one, which is considered as the weights of respective rows. The weights are then multiplied with gradient values of respective rows of the image to widen the gap between pixels in the middle portion of text and the other portions. Next, k-means clustering is performed on WGF to classify middle and other pixels of text. The method performs morphological operation to obtain word components as patches for the result of clustering. The patches in both the clusters are matched to find common patch areas, which helps in reducing touching effect. Then the proposed method checks linearity and non-linearity iteratively based on patch direction to segment text lines. The method is tested on our own and standard datasets, namely, Alaei, ICDAR 2013 robust competition on handwriting context and ICDAR 2015-HTR, to evaluate the performance. Further, the method is compared with the state of art methods to show its effectiveness and usefulness.**

Keywords—Zero crossing points, Gradient values, k-means clustering, Handwritten text line analysis, Text line segmentation.

I. INTRODUCTION

As technology especially devices like smart phone advances the practice of writing and the focus on reduce writing. This results in sloppy, choppy or ugly writing when situation demands writing such as descriptive answers, and filling forms. This makes the segmentation problem complex and cumbersome because such writing introduces severe touching between lines and arbitrary skew. At the same time, line segmentation influences recognition greatly to achieve better and accurate recognition rates [1]. In addition, handwriting recognition is emerged as one of the important applications for studying and predicting human behaviors, characteristics, human interactions and the other forensic applications [2]. Therefore, line segmentation from handwritten document is still considered as challenging and there is a demand for accurate recognition. For example, the sample handwriting image in Fig. 1(a) shows that touching between lines is common for handwriting. It is evident from the Canny edge image of the handwriting document image shown in Fig. 1(a). For such documents, the methods based on projection profile [3, 4, 5] may not segment lines well as shown in Fig. 1(b), while the proposed method segments well for the same document as shown in Fig.1(b).



(a)Handwritten document Canny edge image

(b) Projection profile Proposed Method

Fig.1. Challenges for line segmentation

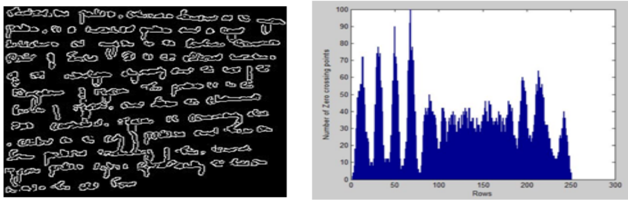
Handwriting segmentation is not a new problem for document analysis community. There are many methods developed in literature to segment text lines from handwritten documents. Alaei et al. [6] proposed a new scheme for unconstrained handwritten text line segmentation. The method introduces a painting idea for enhancing separability between lines. It divides the whole document into stipes and then performs averaging operation to separate foreground information. Though the method is tested on multi-lingual document images, it is sensitive to noises in images and background variations. Sushma and Veena [7] proposed Kannada handwritten line and word segmentation. The method uses binarization and then uses HMM model for segmentation. It is true that binarization is not a good idea for degraded and low-quality documents. Kesiman et al. [8] proposed a new scheme for text line and character segmentation from gray scale images of Palm leaf manuscript. The method uses gray information rather than binarization for segmentation. Horizontal and vertical filters have been used for brushing gray values of text information. Then block projection profiles and non-linear path estimation are used for segmentation. Despite the method does not depend much on binarization, it is sensitive to different skews and touching lines because of block wise projection profiles. Valy et al. [9] proposed a line segmentation approach for Ancient palm leaf manuscripts using competitive learning algorithm. The method uses binarization, connected component analysis, clustering and then learning for segmentation. Since segmentation is a preprocessing step for recognition, the use of a classifier may limit to a specific dataset or application. Vo and Lee [10] proposed dense prediction for text line segmentation in handwritten document images. This method explores deep learning for segmentation. The touching is dealt by line adjacency graph. As mentioned earlier, if the method depends much on a classifier, the number of training samples and parameters, the method may lose generic nature. Therefore, it might not work for images of different scripts. Chavan and Mehrotra [11] proposed a method for text line segmentation of multilingual handwritten documents using Fourier approximation. The method divides the whole page vertically into pieces to find the initial points for text line segmentation. It uses Fourier signals corresponding to the points of projection profiles to detect line separation points, and then uses the line drawing algorithm for text line segmentation. However, it is not clear how this method takes care of severe touching and poor quality documents.

From the above discussion, it is found that most of the methods focus on a particular script but not multi-script documents. In addition, the methods depend much on binarization and learning. Further, the severe touching between lines is not well addressed. Therefore, in this paper, we propose a new idea which does not depend on binarization and is insensitive to scripts, skew and touching. We explore gradient information with the number of zero crossing points to extract the middle portion of text, which in turn helps in reducing touching effect. Then we propose linearity and non-linearity check for eliminating touching, which eventually results in line segmentation. In summary, the main contribution is the way we combine zero crossing points with gradient information for enhancing text pixels. The use of angle information for checking linearity and non-linearity to segment text lines regardless of touching, scripts and orientations is different from literature.

II. Proposed Method

In this work, handwritten documents of different scripts with touching between lines are considered for segmentation. It is a fact that for text lines of any script, one can expect more transitions at the middle portion of text. At the same time, fewer transitions are at the top or bottom portion of text. We explore this observation by counting the number of zero crossing points for every row of the Canny edge image of the input image. It is illustrated in Fig.2, where we can see the highest peaks between two valleys in the histogram represent the middle portion of text area. In the same way, where there is a text pixel, the gradient gives high values and where there is no text pixel, it gives low values. In other words, gradient enhances text pixels by suppressing background pixels. To take the advantage of zero crossing points and the gradient information, we propose to multiply the number of zero crossing points to gradient values of respective rows in the image, which we call Weighted-Gradient Features (WGF). As a result, it is expected a wide gap between text and non-text pixels. To separate text and non-text pixels, we propose to apply K-means clustering with K=2 on the WGF matrix, which outputs two clusters. The cluster, which gives the highest mean, is considered as text cluster, and another as background cluster. In case of a single document containing multiple skewed text lines, one can expect misclassification of texts and background pixels in clustering because weight values do not make much difference when text lines are skewed in multi-directions in a single document. However, gradient value makes difference because it will be high for text pixels compared to background ones irrespective of cursive and skew. We also believe that such documents are seldom when we see applications of handwriting recognition. Therefore, the weight derived using the number of zero crossing points horizontally with gradient value is useful in separating text and background pixels.

Since the middle portion of a text contains high values according to weighted gradient features, the pixels are classified into a high mean cluster, and the other pixels including background ones are classified into a low mean cluster. As a result, one can expect the pixels which cause touching get separated due to the separation of middle and other pixels. In this way, the proposed idea eases the cause of touching. To restore the information in the low mean cluster and the high mean cluster, we perform morphological operation with kernel of size 5×9 to fill the gap between pixels such that each word has become one component. The size of the kernel is determined empirically. Then the proposed method matches the corresponding components in high mean and low mean clusters to choose the common area, which eliminates all the background pixels and even touching also, which are called patches. However, this step alone is not sufficient to remove complete touching, we thus propose linearity and non-linearity checking based on the direction of the patches to eliminate touching, which results in text line segmentation.



Row profiles for Canny edge image using zero crossing points.
Fig.2. Zero crossing points are considered as weights.

A. *Weighted-Gradient Features for Separating Middle Zone from Others in Text Line*

As discussed in the previous section, for a given input image, the proposed method uses Canny edge operator to obtain Canny edge image. It is true that for a plain document image, Canny gives fine edge details compared to the other edge detectors as shown in Fig. 2(a), where we can see most text pixels are retained without introducing any noise pixels. The number of zero crossing points for every row is computed, which are considered as weights. For the same input image, the proposed method obtains a gradient image as shown in Fig. 3(a), where one can see the brightness of edge pixels is increased compared to the pixels in the Canny edge image. This shows that contrast between edge and background pixels is increased. However, this is insufficient to segment lines when a document is affected by touching. Therefore, the proposed method multiplies the number of zero crossing points with gradient values, which is called weighted-gradient image as shown in Fig. 3(b), where we can see the values are enhanced significantly compared to the values in Fig.3(a), especially the pixels which represent the middle portion of text compared to the other portion of pixels. Formally, we can write the steps as follows.

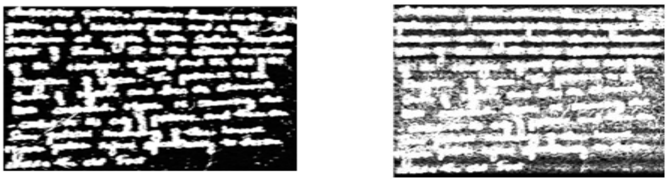
Let G be the gradient image of the input image I of size $m \times n$, then multiplied gradient image, MG can be defined as in equation (1).

$$MG = ZC * G \tag{1}$$

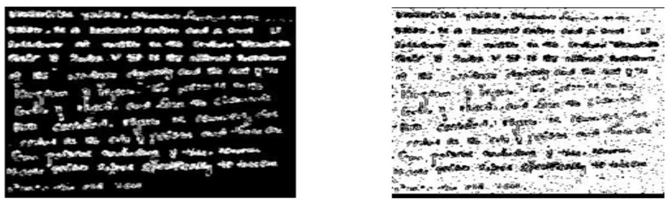
where ZC is zero crossing of each row in image I. The weighted-gradient image is computed as defined in Equation (2).

$$for\ i = 1{:}m\ \{ZC(i) = ZC(i) + \textstyle\sum_j^n I(i,j)\},$$
$$when\ I(i,j) == 1 \tag{2}$$

Since the above step widens the gap between the pixels that represent the middle portion and those that represent the other portion, we propose to employ K-means clustering with K=2 on the weighted-gradient image to classify middle pixels into one cluster, which is nothing but High Mean Cluster (HMC), and the other pixels including background ones are classified into another cluster, which is called Low Mean Cluster (LMC) as defined in Equation (3). The effect of K-means clustering is illustrated in Fig.3(b), where it can be seen that HMC loses top and bottom pixels of text components, at the same time, LMC gets missed pixels as we can see white pixels in LMC. Interestingly, compared to the results in Fig. 3(a), the results in HMC and LMC get fewer number of touching. This is the advantage of weighted gradient features.



(a) Gradient image and weighted gradient image for the input image(a)



(b) High mean cluster (HMC) Low mean cluster (LMC)

Fig.3. Separation of middle portion pixels from other portion of text.

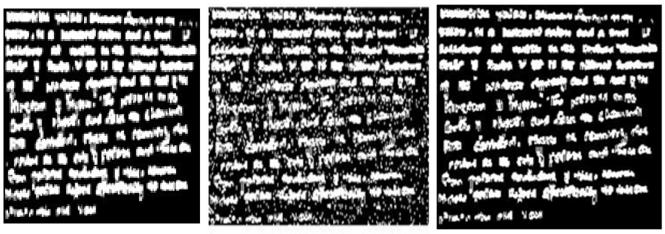
$$Kmeans(MG, 2) = \begin{cases} HMC & Middle\ Pixels \\ LMC & Other\ Pixels \end{cases} \tag{3}$$

B. Linearity and Non-Linearity Check for Line Segmentation

For the results given by the previous step, we perform morphological operation to fill the single pixel gap to avoid disconnections as shown in Fig. 4(a) and Fig. 4(b), where MHMC denotes the result of morphological information over HMC, and MLMC denotes the results morphological operation over LMC. In order to restore the missing text information in HMC, we perform inversion operation over MLMC such that text components will be represented by white pixels, and background will be represented by black pixels as MHMC. The effect can be seen in Fig. 4(b). This results in patches. Since we use gradient information of input image for clustering, a few background pixels are misclassified as text. To remove such noisy pixels and restore actual text patches, we propose to compare the boundary of the patches in the MLMC with the patches of MHMC to find the common area of patches as defined in Equation (4). Note that it is not pixel by pixel comparison to find the common area. We believe that the Common Patches (CP represents the actual text information). Therefore, this operation helps us to eliminate noisy background pixels as well as touching between lines as shown in Fig.4(c). However, it is noticed from Fig.4(c) that still touching exists.

$$CP = \begin{cases} intersection\big(cc_{HMC}(i), cc_{LMC}(i)\big) \\ + \\ boundaryMatching\big(cc_{HMC}(i), cc_{LMC}(i)\big) \end{cases} \tag{4}$$

where CP denotes Common Patches, $cc_{HMC}(i)$ is the i^{th} component of *MHMC* and $cc_{LMC}(i)$ is the i^{th} component of *MLMC*.

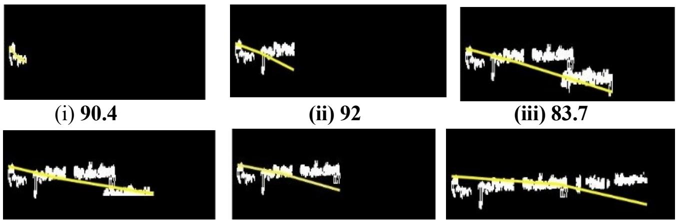


(a) MHMC (b) MLMC (c) Common patches.

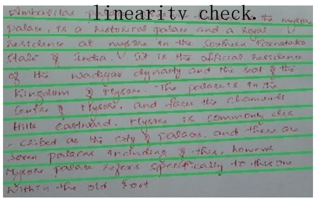
Fig.4: Matching boundary of the components to find common patches

To eliminate such touching, the proposed method introduces linearity and non-linearity checking based on the directions of patches. For each patch in the common patch image, the proposed method finds a direction by Principal Component Analysis (PCA), which gives the principal axis for each component in the image. Then the method moves in the same direction of the component until it gets the nearest neighbor component. The angle of the first component and the neighbor component are compared. If the absolute difference is more than 2º, it is suspected touching else the neighbor component will be merged with the first component. If there is a touching, the method removes bottom pixels of the neighbor component, which is to be merged iteratively pixel by pixel. At the same time, for each iteration, the method compares the angle between the first and its neighbor component. This process continues until the difference becomes less than one degree. In this way, the proposed method overcomes the problem of touching to segment text lines.

The whole process is illustrated in Fig. 5(a), where (i) shows the first component that gives 90.4 degree, which is iteration-1. The method moves in the same direction until it touches a neighbor component as shown in (ii), where the first and the second components together give 92 degree, which is itration-2. The method finds the absolute difference between the angles of the first and the second iterations. When the difference is less than 2º, it is considered as linear. The value of 2º is determined empirically here. The process continues to find neighbor components for the component given by iteration-2 with the same direction as shown in (iii), where 83.7 is given. Since the difference between the angles of itration-2 and iteration-3 is more than 2º, it is said to be non-linear. This indicates there is a touching between text lines. To overcome this problem, the method removes bottom pixels of the third component gradually until it gives the angle which satisfies the linearity condition as shown in (iv) and (v), where the angle difference between (ii) and (v) is less than 2º, which satisfies the linearity condition. In this way, the iterative process continues until the end of the line as shown in (vi). The effect of linearity and non-linearity check for the input image affected by touching is shown in Fig. 5(b), where all the lines are segmented well.



(a) Illustrations for iterative linearity and non-linearity check.



(b) Final segmentation results.

Fig.5. Line segmentation by iterative linearity and non-linearity check.

III. EXPERIMENTAL RESULTS

The primary aim of the proposed work is to segment text lines from multi-script documents especially Indian scripts because we can expect more touching for Indian scripts compared to English and the other scripts. This is due to the more cursive nature of Indian scripts. Therefore, we create our dataset of 144 handwritten document images, which includes English and Indian scripts, namely, Hindi, Kannada, Telugu, Malayalam and Tamil. In addition, the dataset comprises images of different writers, genders, pens, ink, papers, etc. To test the effectiveness of the proposed method, we conduct experiments on three benchmark datasets, namely, Alaei et al. [12] which contains 200 Bangla, 228 Kannada and 140 Oriya handwritten images, ICDAR 2013 handwriting context competition [13] which provides 350 handwritten images, and ICDAR 2015-HTR [14] handwriting recognition context which provides 746 handwritten images. In total, 1826 (144+586+350+746) handwritten images are considered for evaluating the proposed segmentation method in this work. ICDAR 2015-HTR dataset [14] is available for recognizing context but not line segmentation. However, we prefer to use this dataset for evaluating the proposed method on line segmentation. This is because the dataset is collected from Bentham collection used in the TRANSCRIPTORIUM project. The dataset is written by several hands and entails significant variability and difficulties regarding the quality of text images, writing style and crossed out text. Since the ground truth for ICDAR 2013 and ICDAR 2015 data is available, we use the same ground truth for calculating measures. While for the other two datasets, we count manually since the dataset does not provide ground truth.

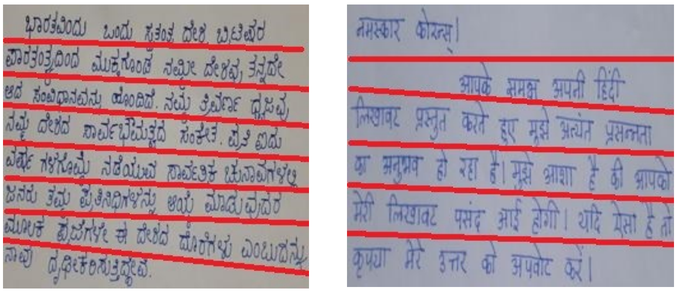
To calculate measures, we use the standard evaluation scheme proposed in ICDAR 2013 handwriting segmentation contest [13], where they defined the following measures. Let *I* be the set of all the image points, *Gj* be the set of all the points inside the *j* ground truth region, *Ri* be the set of all the points inside the *i* result region, and *T(s)* be the function that counts the elements of set *s*. *MatchScore*(*i,j*) represents the matching results of the *j* ground truth region and the *i* result region:

$$MatchSscore(i,j) = \frac{T(G_j \cap R_i \cap I)}{T((G_j \cup R_i) \cap I)} \qquad (5)$$

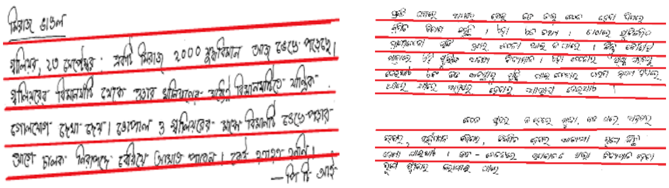
We consider a region pair as a one-to-one match only if the matching score is equal to or above the acceptance threshold *Ta* of the evaluator. The acceptance threshold is used as *Ta*=95% according to ICDAR 2013 [13]. If *N* is the count of ground-truth elements, *M* is the count of result elements, and o2o is the number of one-to-one matches, we calculate the Detection Rate (DR) and Recognition Accuracy (RA) as follows: a performance metric *FM* can be computed if we combine the values of detection rate and recognition accuracy as defined in Equation (6).

$$DR = \frac{o2o}{N}, \quad RA = \frac{o2o}{M}, \quad FM = \frac{2\,DR\,RA}{DR+RA} \qquad (6)$$

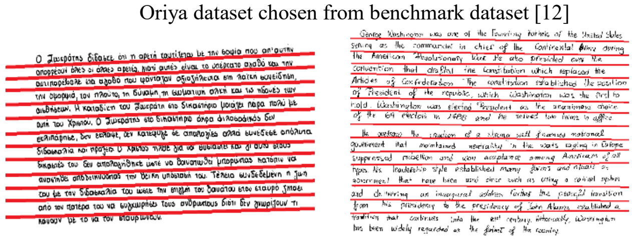
To show the effectiveness of the proposed method, we implement two state of the art methods. Valy et al. [9] proposed lien segmentation approach for ancient palm leaf manuscripts using competitive learning algorithm, which works based on piece wise projection profile analysis and connected component analysis. Kesiman et al. [8] proposed a new scheme for text line and character segmentation from gray scale image of palm leaf manuscript. This method also explores projection profile analysis and non-linear path reconstruction for segmentation. Since the codes of the above existing methods are not available, we implement them according to the instructions in the papers.



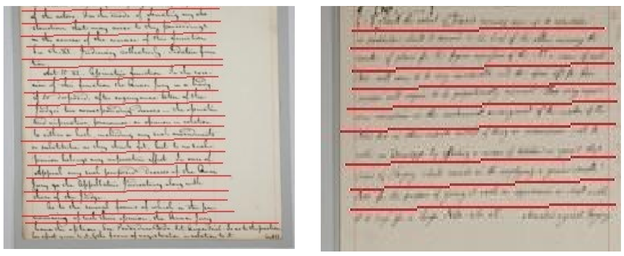
(a) Line segmentation of the proposed method on Kannada and Hindi chosen from our dataset.



(b) Line segmentation of the proposed method on Bangla and Oriya dataset chosen from benchmark dataset [12]



(c) Line segmentation of the proposed method ICDAR 2013 Line Segmentation dataset [13]



(d) Line segmentation of the proposed method on English chosen from Bentham collection (ICDAR 2015 HTR context [14]).
Fig.6. Sample line segmentation results of the proposed method on different datasets.

Sample qualitative results of the propose method for the four datasets are shown in Fig.6(a)-Fig.6(d), where it is noticed that the proposed method segments lines well for the images of our dataset, Alaei, ICDAR 2013 and ICDAR 2015 datasets, respectively. This shows that the proposed method is independent of scripts and works well for low contrast, low quality documents and the documents that have touching. Quantitative results of the proposed and existing methods for the four datasets are reported in Table I, where it is noted that the proposed method achieves the best results in terms of detection rate, recognition accuracy and the overall-segmentation accuracy compared to the existing methods. The reason for the poor results of the existing methods is that since the method depends on binarization, projection profile analysis and connected component analysis, the methods are not robust to low quality, different scripts and touching. On the other hand, the proposed method involves new robust weighted-gradient features and linearity-non-linearity checking, it is better than the existing methods

Table I. Performance of the proposed and existing methods for line segmentation on different datasets

Methods / DataSet	Proposed Method			Valy et al.[9]			Kesiman et al.[8]		
	DR	RA	FM	DR	RA	FM	DR	RA	FM
Our dataset	**98.3**	**97.4**	**97.7**	95.5	94.1	95.3	92.1	90.4	91.3
Alaei [12]	**97.8**	**96.7**	**97.1**	95.8	94.6	95.2	91.6	90.6	91.6
ICDAR2013 [13]	**96.4**	**94.3**	**94.8**	93.7	92.2	92.9	90.1	90.8	90.7
ICDAR2015-HRT [14]	**87.6**	**86.8**	**87.2**	86.6	86.6	86.6	77.5	77.2	77.4

When the FM score of the proposed method is compared with the score (98.66%) of the best method, INMC in ICDAR 2013 handwriting segmentation competition and the score (98.16%) of the latest method [11], there is a marginal difference 3.86%. We believe that such marginal difference does not make much difference for recognition. Besides, since the proposed method focuses on text line segmentation from South Indian scripts, which are more cursive than Bangla, Greek, Spanish and English, sometimes, the proposed PCA for linearity and non-linearity check misses the actual direction of text lines especially when there is severe touching. Further, the scope of the proposed method is to address documents which suffer from aging and poor quality. Therefore, overall, the proposed method is competitive and useful compared to the state-of-the-art methods. It is observed from Table I that the proposed and existing methods score poor results for the ICDAR 2015-HTR dataset compared to those of the other three datasets. This is because ICDAR 2015-HTR dataset is much more complex, which suffers from poor quality, though it does not include multiple scripts images. As mentioned-above, the proposed method fails to segment text lines properly for the poor quality documents as shown in Fig. 7. Here the proposed method misses a few text lines. This is due to the limitation of the proposed step called linearity and non-linearity check using PCA. When a component loses pixels and shape, PCA sometimes fails to predict the correct direction.

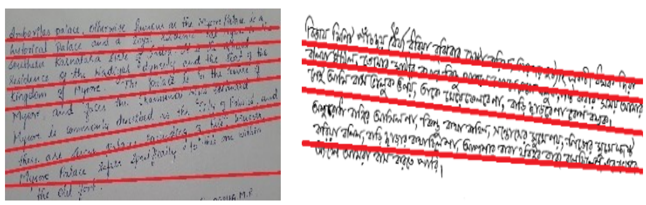


Fig.7: The proposed method fails when PCA of linearity and non-linearity check does not give correct direction of text line.

IV. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a new method for segmenting text lines from handwritten documents. The proposed method introduces new features, which integrate the advantage of zero crossing points and gradient information to enhance text pixel information by suppressing background information. K-means clustering is used for separating middle pixels from other pixels in enhanced images, which results in two clusters, namely, high mean cluster and low mean cluster. We propose matching criteria for the patches in the two clusters at boundary level to find common patches. For the common patches, the proposed method further explores linearity and non-linearity checking to segment text lines irrespective of touching between lines. It is noted from Table I that the proposed and existing methods report poor results on ICDAR 2015 HTR dataset. Therefore, we have planned to extend the same idea for improving the results for poor quality images in the near future. Besides, sometimes, when the first component gives incorrect angle, the method does not perform well. Therefore, we will make the method invariant to starting patches in the next work.

ACKNOWLEDGMENT

The work described in this paper was supported by the Natural Science Foundation of China under Grant No. 61672273 and No. 61272218, and the Science Foundation for Distinguished Young Scholars of Jiangsu under Grant No. BK20160021, and Scientific Foundation of State Grid Corporation of China (Research on Ice-wind Disaster Feature Recognition and Prediction by Few-shot Machine Learning in Transmission Lines).

REFERENCES

[1] V. Romero, J. A. Sanchez. V. Bosch, K. Depuydt and J. D. Does, “Influence of text line segmentation in handwritten text recognition”, In Proc. ICDAR, pp 536-540, 2015.

[2] S. He and L. Schomaker, “Beyond OCR: Multi-faceted understanding of handwritten document characteristics”, Pattern Recognition, 63, pp 321-333, 2017.

[3] S. Eskenazi, P. G. Kramer and J. M. Ogier, “A comprehensive survey of mostly textual document segmentation algorithms since 2008”, Patten Recognition, 64, pp 1-14, 2017.

[4] V. Papavassiliou, T. Stafylakis, V. Katsouros and G. Carayannis, “Handwriting document image segmentation into text lines and words”, Patten Recognition, 43, pp 369-377, 2010.

[5] G. A. Farulla. N. Murru and R. Rossini, “A fuzzy approach to segment touching characters”, Expert Systems with Applications, 88, pp 1-13, 2017.

[6] A. Alaei, U. Pal and P. Naghabushan, “A new scheme for unconstrained handwritten text-line segmentation”, Pattern Recognition, 44, pp 917-928, 2011.

[7] A. Sushma and G. S. Veena, “Kannada handwritten word conversion to electronic textual format using HMM model”, In Proc. ICCSISSS, pp 330-335, 2016.

[8] M. W. A. Kesiman, J. C. Burie and J. M. Ogier, "A New Scheme for Text Line and Character Segmentation from Gray Scale Images of Palm Leaf Manuscript”, In Proc. ICFHR, pp. 325-330, 2016.

[9] [9] D. Valy, M. Verleysen and K. Sok, "Line Segmentation Approach for Ancient Palm Leaf Manuscripts Using Competitive Learning Algorithm", In Proc. ICFHR, pp. 108-113, 2016.

[10] [10] Q. N. Vo and G. Lee, “Dense prediction for text line segmentation in handwritten document images”, In Proc. ICIP, pp 3264-3268, 2016.

[11] V. Chavan and K. Mehrotra, “Text line segmentation of multilingual handwritten documents using Fourier approximation”, In Proc. ICIP, 2017, pp 250-255.

[12] A. Alaei, U. Pal and P. Nagaghushan, “Dataset and ground truth for handwritten text in four different scripts”, IJPRAI, Vol. 26, pp. 1253001-1253025, 2012.

[13] N. Stamatopoulos, B. Gatos, G. Louloudis, U. Pal and A. Alaei, “ICDAR 2013 handwriting segmentation contest”, In Proc. ICDAR, 1402-1406, 2013.

[14] J. A. Sanchez, A. H. Toselli, V. Romero and E. Vidal, “ICDAR 2015 competition HTRtS: Handwritten text recognition on the transscrptorium dataset”, In Proc. ICDAR, pp 1166-1170, 2015.