

Skewed Line Detection and Removal Preserving Handwritten Strokes: A New Approach

Amjad Rehman ^{#1}, Dzul kifli Mohammad ^{#2}, Tanzila Saba ^{#3}

[#]Department of Computer Graphics and Multimedia

University Technology Malaysia 81310 Skudai Johor Malaysia

¹amjadbzu2003@yahoo.com

²dzulkifli@utm.my

Abstract— Text overlapping with lines poses serious problems for the optical character recognition systems. The dilemma becomes crucial for skewed and non-uniform thick line present in the word image. Although detection and removal of the straight underlines has been addressed but still skewed lines removal and restoration of the area after removal of lines persists to be a problem of interest. A new method is proposed to detect and remove skewed and straight line at any position inherited in the word image without characters distortion to avoid restoration stage by preserving strokes. The proposed technique is based on connected component analysis and is equally suitable to remove straight and skewed line from printed and handwritten words. Detailed experiments are conducted on manually filled forms of National Institute of Standard and Technology (NIST) special benchmark database19. Comparisons with other methods available in the literature exhibit potential of the new approach with accuracy up to 95.18%.

Keywords— underline detection and removal, character restoration, connected component analysis, handwriting recognition, image analysis.

I. INTRODUCTION AND BACKGROUND

In document processing as well as some other applications such as invoices, lines that interfere with text present a significant problem for OCR systems [1]. Even a small misalignment against pre-printed forms can result in a majority of the text on a page being partially obscured by horizontal or vertical lines. One of the hardest problems facing

line removal algorithms is how to remove all the pixels in a line without removing semantically significant blobs that are morphologically connected [2]. The problem is more controversial for hand printed skewed and non-uniform thick lines. The detection and removal of these factors through preprocessing techniques can be helpful to reduce variability and to improve recognition rates [4].

Literature exhibits several approaches for underline detection and removal in text. Most of them detached underline from binarized image by the dilation and erosion operators of the mathematical morphology [2, 3, 5, 6]. Dilation was applied until all the lines longer than a fixed threshold are removed from the underline region. On the other hand this operator shattered the characters and therefore it became difficult to recognize. Hence erosion was applied to recover the lost parts of the characters. However, broken characters could not restore correctly [7]. In the same way, Dimauro et al. in [8] performed an experimental investigation on algorithms proposed in [5] and [7] for underline removal based on mathematical morphology and finally, proposed a new approach based on dynamic selection of the structuring element. Although their system performs well, it does not seem to take into account the possibility of skewed handwriting. In some algorithms such as proposed in [8], broken characters are restored. If result that character recognizer is performed with the restored characters is wrong, restored characters are sent back to the restoration algorithm stage. In such methods, the processing time was increased because it has feedback paths. In addition, characters are sometimes recognized incorrectly such as ‘h’ and

'b' [3]. Govindaraju and Srihari [9] achieved underline removal by using the "good continuity criterion". The criterion first detects the smooth strokes in the image and then identifies the spine of the image as the smooth stroke with maximum length and finally is removed. Unfortunately, this approach works only on thinned images and therefore it requires a preliminary time consuming process. Blumenstein et al., [4] introduced new preprocessing techniques for underline removal based on horizontal black pixel runs analysis based on two assumptions. Firstly, It was assumed that word stroke thickness will be similar to the thickness of the underlines present in the image. However this assumption is not true in all cases particularly for printed documents/forms. Secondly, text and line have the same skew angle. Beside that, Blumenstein et al., [4] agreed that underline removal did not perform well on some of the more difficult erratic and skewed underlines that were present in some word images. Therefore remainders of undetected underlines were removed manually to facilitate further processing.

Recently, Arvind et al., [10] detect multiple printed lines with varying thickness present in the word image using horizontal projection profile. Restoration of the smashed characters is performed by using Bresenham line drawing algorithm [11]. However, technique can not deal with restoration of printed characters and skewed images.

In this paper, we have two challenges: 1) the hardest problem of skewed line removal, and 2) preserving the handwritten strokes to skip restoration stage. By using a combination of thickness and connected component analysis, we have created an effective solution to this problem. The rest of the paper consists of four parts: analysis of connected components, implementation and experimental results, analysis and comparison of results and finally conclusion is drawn.

II. CONNECTED COMPONENT ANALYSIS

The precise analysis is a prerequisite to detect unwanted line and junctions. A junction point is defined as a contact point of characters with line as shown in Figure 1. In implementation the line is neither a correct straight line nor of uniform

thickness and its position is random. Before removal of detected line, junctions are examined at each pixel to avoid characters smash up.



Fig. 1: Line and junction points

III. IMPLEMENTATION OF THE PROPOSED ALGORITHM

A. Overview of the proposed approach

The proposed approach is based on connected components analysis. The algorithm consists of two main modules. Skewed / straight line detection and checking of junction points prior to removal of detected line to avoid character strokes distortion.

1) *Line detection*: Foremost, to detect unwanted line, procedure starts by tracing left most foreground pixel. The traced component is analysed by checking connected pixels (without drastic change) from left to right along with record of connected components length to save "t/T", "J" bar. Connected components are considered line if its length is greater than half of the width of word.

2) *Line removal and junction detection*: Prior to removal of the detected line, junctions are examined by tracing pixels up and down to some threshold in order to avoid character distortion. Finally, connected components of the detected line are simply converted from foreground to background pixels except at junction points to avoid characters distortion.

B. Proposed algorithm

1) Line detection module

Let say an image denoted by P where $P \in \{0,1\}$, (1 represent foreground pixel)

$$p_{i,j} \in P, \begin{cases} i = 1, 2, \dots, h \\ j = 1, 2, \dots, w \end{cases}$$

h, w height and width of P respectively.

Define origin as $O = \{o \in P | o = 1\}$. Take origin point $o_{a,b} \in O$, a and b is left most of P .

(i) Define

$L = \{l \in \{p_{i,j}\} \mid p_{i,j} = 1, p_{i,j+1} = 1, j = 1, 2, \dots, a, a \leq w\}$ Start from $o_{a,b}$ trace line (L) to right direction allowing one pixel upward and downward continually.

(ii) Calculate $length(L) = \begin{cases} length(L)+1, p_{i,j} = 1 \\ length(L), p_{i,j} = 0 \end{cases}$

(iii) If $length(L) < \frac{1}{4}w$ then do step (i).

2) Line removal module

Connected components of the detected line are simply converted from foreground to background pixels except at junction points to avoid restoration stage. Checking of junction up, down and diagonal is limited by threshold that is calculated from thickness of the line.

$$U = \left\{ u_k \in \{p_{i,j}\} \mid p_{i+r,j+r} = 1, -\left\lfloor \frac{t}{2} \right\rfloor \leq r \leq \left\lfloor \frac{t}{2} \right\rfloor, k = 1..t \right\}$$

If $p_{i,j} \in L$ and $\begin{cases} p_{i-1,j} \neq 1 \\ p_{i+1,j} \neq 1 \end{cases}$ and

$length(U) < thick$ then $p_{i,j} = 0$ (changing foreground to background)

IV. PLATFORM ENVIRONMENT AND EXPERIMENTAL RESULTS

NIST SD19 [15] is used for the experiments that contain 3699 handwriting sample forms with 34 fields each scanned at 300dpi. For meaningful experiments, 2785 word images overlapped with line (skewed/straight) are extracted. Following line removal, results are divided into four categories A, B, C and D. Out of 2785 word images containing some form of line, 2651 were cleaned (95.18 %) by the new method. Table 1 presents line removal results in four categories. Successful results are presented in figure 2 while failure results are reported in figure 3.

A: Very good: Line completely removed without loss of information.

B: Noisy: Line not completely removed or noise in the final image

C: Eroded: Line removed with partial loss of information

D: Not acceptable: Image very noisy and/or the line have been removed with significant loss of information.

Table 1. Benchmark test results for images overlapped with some form of lines

| Type of line | Samples | Results | | | |
|---------------|---------|---------|----|----|----|
| | | A | B | C | D |
| Straight line | 1849 | 1782 | 39 | 17 | 11 |
| Skewed line | 936 | 869 | 27 | 31 | 9 |

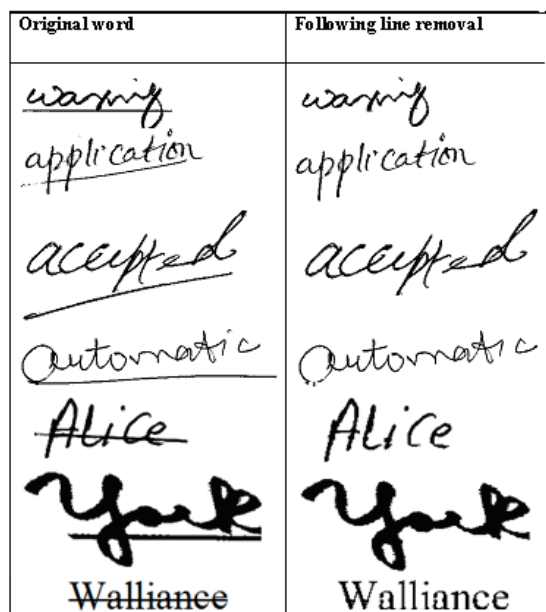


Fig 2: Words samples before and after line removal

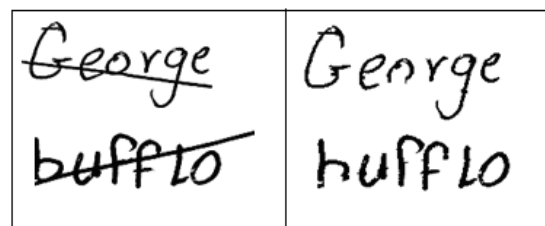


Fig 3: Failure results for skewed line removal

V. ANALYSIS AND COMPARISON OF RESULTS

This section is devoted to exhibit an analysis and comparative study of proposed algorithm discussed in section 3. This study has been made after performing various experiments on handwritten words containing some form of overlapped lines from NIST SD 19 [15] benchmark database.

However, there is no standard to examine accuracy of preprocessing techniques except by our eyes even then opinions may differ [12]. Analyzing failure results, it was found that some noise left with those characters containing long horizontal strokes such as “t”, “h”, “f” etc. Secondly slanted characters also created minor errors. Finally, it is very difficult, if not impossible to distinguish without contextual information that part of lines just overlapped with handwritten strokes as shown in figure 3.

Finally, it was also hard to compare results for line removal as most of the researchers used subjective evaluation. The comparisons presented below have been chosen as the results are some of the most recent in the literature.

Blumenstein et al., [4] claim underline removal accuracy up to 97.8% but it does not seem to deal with real skewed line removal. Bai and Huo [14] assert 98.4 % and 94.4 % accuracy for untouched and touched underline removal from printed text taken from UWI [13]. However, they dealt with underlines detection and removal in printed text only. Recently, Arvind et al., [10] affirm line detection and removal accuracy of 86.33% for subjective evaluation. Yet the approach failed in case of line removal from printed text (see sample 2 fig. 4), moreover, it could not deal with skewed line removal. Figure 4 exhibits potential of new proposed approach in comparison with Arvind et al., [10]. The new approach detects and removes line from script writing and printed text while preserving strokes and therefore no restoration stage is applied that reduce processing and increase speed. Whereas, in [10] restoration of broken characters was performed using Bresenham line drawing algorithm [11].

To summarize, unlike [1-5], line removal approach does not damage the characters that eliminate broken character restoration stage. Contrary to Dimauro et al., [1], skewed line removal is possible and is independent from word

slope angle, stroke thickness independence contrasting to [4]. In distinction to [10] proposed approach is equally suitable for line removal from printed and handwritten text. Lastly, the proposed approach is independent from restoration stage as it preserves strokes.

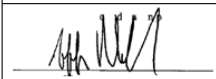
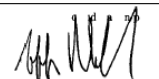
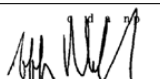
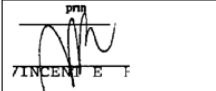
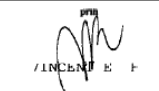
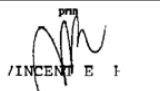
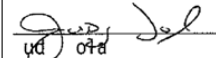
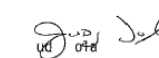
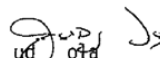
| Handwritten and printed text with an unwanted line | Arvind et al.,[10] | Proposed approach |
|--|---|---|
|  |  |  |
|  |  |  |
|  |  |  |

Fig 4: Results for line detection and removal in handwritten / printed text are compared

VI. CONCLUSION

In this paper, a novel algorithm for line removal regardless of its slope, position and thickness is proposed that preserve handwritten strokes to skip restoration stage. A preliminary comparison of proposed approach with others available in the literature is also presented. The proposed approach is objective and comprehensive. Connected component analysis based algorithm is equally suitable for skewed / straight line removal from handwritten and printed text without character distortion. Moreover, it relatively save computation time and improve accuracy. The evaluation is based on detailed experiments carried out on wide range of noised images in NIST database and found that it is accurate within all practical limits. Further we are working to upgrade the approach to detect and remove vertical lines in script writing preserving handwritten strokes.

REFERENCES

- [1] G. Dimauro, S. Impedovo, G. Pirlo and A. Salzo, “Removing Underlines from Handwritten Text: An Experimental Investigation”, Progress in Handwriting Recognition, A. C. Downton & S. Impedovo, World Scientific Publishing, pp. 497-501, 1997.

- [2] L.C. Sim, H. Schroder and G. Leedham, "Fast line detection using major line removal morphological Hough transform. Proceeding of the 9th international conference on neural information processing, vol. 4, pp. 2127-2131, 2002.
- [3] J. Yong, M. K. Kim, S. W. Bana, and Y. B. Kwon, "Line Removal and Restoration of Handwritten Characters on the Form Documents". Proceedings of the Fourth International Conference on Document Analysis and Recognition, vol. 1, pp. 128-131, Aug 18-20, 1997.
- [4] M. Blumenstein, C.K. Cheng, and X. Y. Liu. "New Preprocessing Techniques for Handwritten Word Recognition". Proceedings of 2nd International Conference on Visualization, Imaging and Image Processing, ACTA. Press, Calgary, 480-484, 2002.
- [5] J. Serra, "Image Analysis and Mathematical Morphology", Academic Press, London. 1982.
- [6] R. Charles, Giardina, R. Edward and Dougherty, "Morphological Methods in Image and Signal Processing", Prentice Hall, Inc. page 264-279, 1988.
- [7] D. Guillevic, and C.Y. Suen, "Cursive Script Recognition: A Fast Reader Scheme". Proceedings of the 3rd International Conference on Documents Analysis and Recognition, 311-314, 1993.
- [8] D. Wang, and S.N. Srihari, "Analysis of Form Images". Proceeding of the International Conference on Document Analysis and Recognition, 181-186, 1991.
- [9] V. Govindaraju, and S.H. Srihari, "Separating Handwritten Text from Interfering Strokes, From Pixels to Features III-Frontiers in Handwriting Recognition, S. Impedovo, J.C. Simon (eds.), North-Holland Publication, 17-28, 1992.
- [10] K.R. Arvind, J. Kumar, and A. G. Ramakrishna, "Line Removal and Restoration of Handwritten Strokes". Proceeding of International Conference on Computational Intelligence and Multimedia Applications, pp. 208-214. 2007
- [11] J. D. Foley, A. V. Dam, S. K. Feiner, and J. F. Hughes, "Computer Graphics: Principles and Practice in C", 2nd Edition, Addison-Wesley, Pearson Education
- [12] E. Kavallieratou, N. Fakotakis, and G. Kokkinakis, "A Slant Removal Algorithm. Pattern Recognition", vol. 33(7), pp. 1261-1262, 2000.
- [13] S. Chen, M. Y. Jaisimha, J. Ha, R. M. Haralick, and I. T. Phillips, "Reference Manual for UW English Document Image Database I," University of Washington, August, 1993.
- [14] Z-L. Bai and Q. Huo, "Underline detection and removal in a document image using multiple strategies. Proceedings of the 17th International Conference on Pattern Recognition (ICPR04), vol. 2, 578- 581, 2004.
- [15] P. J. Grother, NIST Special Database 19-handprinted forms and characters database. National Institute of Standards and Technology, March 1995.