Ruling Line Detection and Removal

Ergina Kavallieratou[a] Daniel Lopresti[b] and Jin Chen[b*]

[a] University of the Aegean, Karlovassi, Samos, Greece;
[b] Lehigh University, Bethlehem, PA 18015, USA

ABSTRACT

In this paper we present a procedure for removing ruling lines from a handwritten document image that does not require any preprocessing or postprocessing tasks and it does not break existing characters. We take advantage of common ruling line properties such as uniform width, predictable spacing, position vs. text, etc. The deletion procedure of the detected ruling line is based on the fact that the coordinates of three collinear points have a determinant equal to zero. The system is evaluated on synthetic page images in five different languages and is compared to a previous methodology.

Keywords: Document Images, Ruling Lines Detection, Ruling Lines Removal

1. INTRODUCTION

Line processing is a necessary task in many systems, including graphic/text discrimination [1], form or invoice processing [2-3], and engineering drawings [4]. Among these, handwritten documents are a special case where ruling lines are used as guides to make it easier to write neatly. Such lines generally share some common characteristics:

1. They are uniform in thickness.
2. Their positions are more-or-less predictable on the page.
3. They are lighter in color and thickness than the handwritten text.
4. Even careful writers often overlap ruling lines.

In previous work, Abd-Almageed et al. [5] introduced a page rule line removal algorithm based on linear subspaces. During a training phase, they incrementally construct linear subspaces representing horizontal and vertical lines using a set of rule line-only images. During the testing phase, they measure the distance between features extracted from the test image and the previously constructed subspace. To identify rule line pixels, the feature vector is projected onto the subspace model and the reconstruction error is computed. If the error is larger than an experimentally determined threshold, the pixels are considered foreground; otherwise they are ruling line pixels. They also introduce a scheme for evaluating noise removal. Their subspace method achieves approximately 88% for both recall and precision on 50 test images of Arabic handwriting.

Arvind et al. [6] proposed a method where the document image is cleaned of noise, segmented into blocks, and skew-corrected. Ruling lines are detected and then removed based on a run-length analysis. Finally, overlapping handwritten characters are repaired by detecting strokes and filling in the missing area. They quote a subjective evaluation with an accuracy of 86.33%.

Our team has also presented a methodology for ruling line removal in the past [8]. However, a disadvantage of this earlier technique is the need for pre- and post-processing noise removal. Moreover, the method is less robust in the presence of skew.

In this paper, we present a completely new methodology for line detection and removal capable of eliminating ruling lines from a page image without damaging the text. It does not require any pre-processing or post-processing stages.

[*] Further author information: (Send correspondence to E.K.)
E.K.: E-mail: kavallieratou@aegean.gr Telephone: +30 6977931514
D.L.: E-mail: lopresti@cse.lehigh.edu
J.C.: E-mail: jic207@cse.lehigh.edu

Moreover, it does not need training. Since it detects the presence of a rulling line before removing it, it has no effect on documents without ruling lines. Finally, it can be applied to both horizontal and vertical ruling lines.

A description of the proposed system follows in Section 2, while experimental results and a discussion of future work are presented in Sections 3 and 4, respectively.

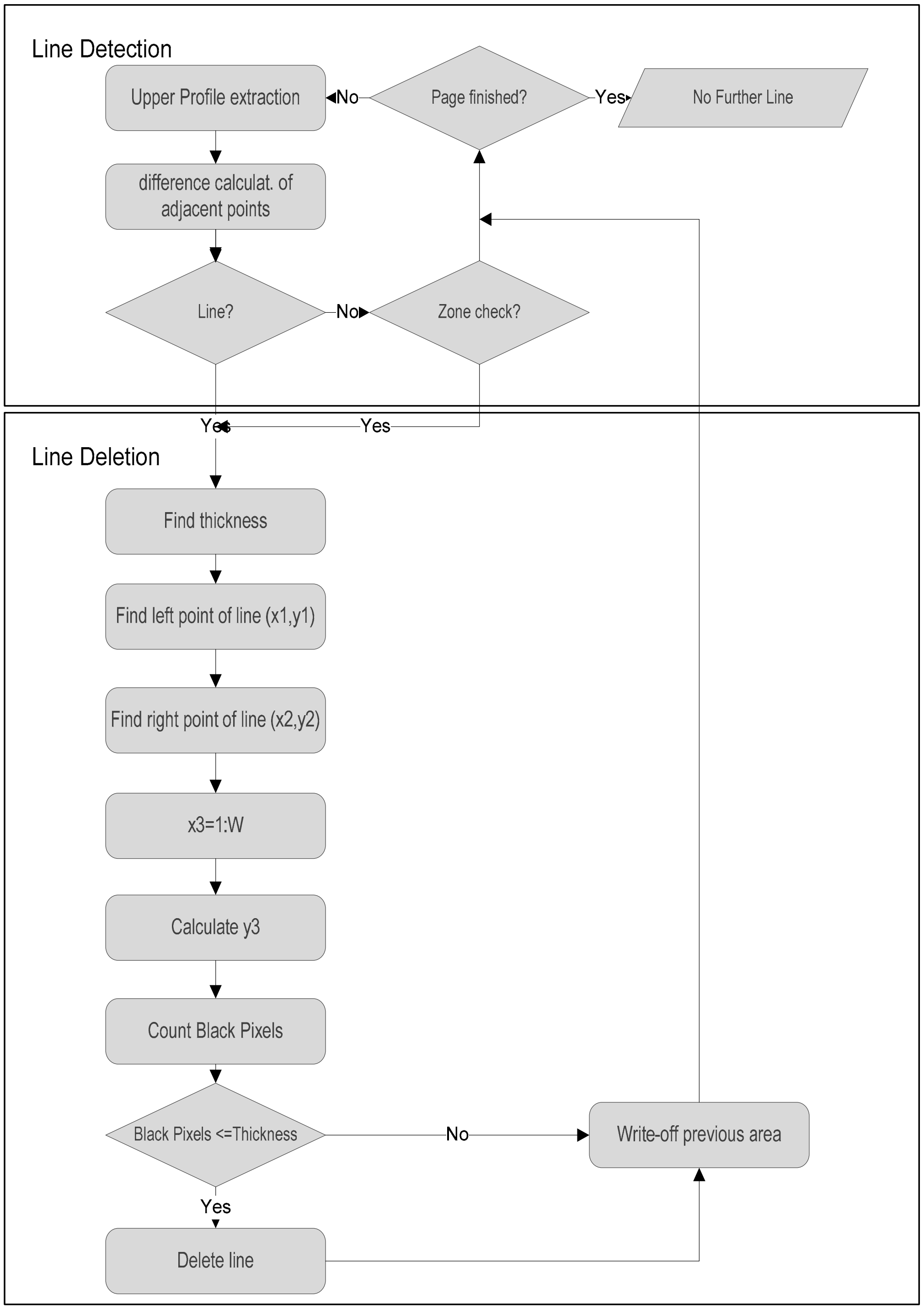

Figure 1. The Proposed System.

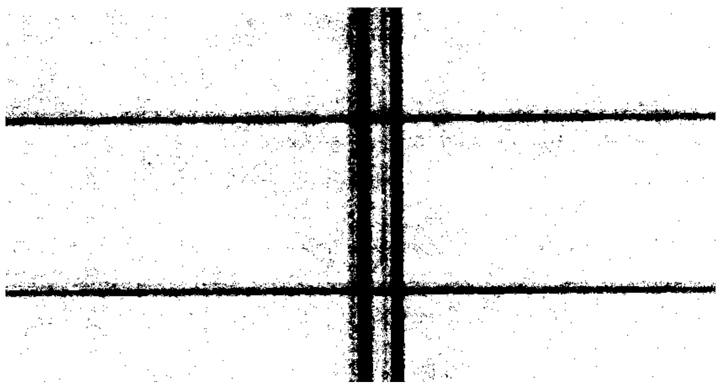

Figure 2. Detail of document image, scanned in grayscale and high resolution and then binarised.

2. SYSTEM DESCRIPTION

The proposed procedure is shown in Fig.1. It consists of two main subtasks: line detection, where the page is examined to find out if there are ruling lines, and line deletion, where if a line has been detected, it is carefully examined in order to be deleted. These two subtasks are described more detailed next.

2.1 Line Detection

In an image document processing system, it is important to know in advance if there are ruling lines in the page in order to avoid affecting pages without ruling lines. In our case this decision is based on the upper profile of the page. The upper profile consists of the first black pixels, met on each column of the image, starting from the top. If there are ruling lines in the page, the upper profile should include many points of the upper ruling line. Even in the presence of text many points of the upper ruling line, before and after the text or between the characters, will be included in the upper profile.

In this case, the adjacent pixels of the ruling line that are included in the page would differ vertically by 0 (no or little skewed page) or 1 (skewed page). In case that there are no ruling lines in the page, between the characters would appear deep differences in the profile. Thus if the majority of the profile points differ vertically 0 or 1, it means that a ruling line exists. In our system, the distances for all the adjacent pixels of the upper profile is calculated and it is checked if the amount of points that differ 0 or 1 are more that W/n, where W is the width of page and n a value that determines the rate of 0 or 1 differences. It depends on the amount of text or/and on the presence of broken ruling lines in the page. As a possible ruling line position is considered the position, where the most differences of 0s and 1s are met. In our case, for the experimental results of section 3, we used n=4.

Another difficult case of document images is those that have been scanned in grayscale and high resolution and then they are binarised. In this case the limits between black and white are not very clear (Fig.2). Thus, our methodology could fail to conclude by examining the difference of adjacent pixels. In order to cover these cases, if the standard procedure, described above, reaches a negative decision cause to not enough (in number) differences of 1 and 0, a zone around the possible ruling line position, described above, is checked and if black pixels at all columns of the zone are met, the decision is changed to positive. As you understand this technique does not permit broken ruling lines in the case of document page scanned in grayscale high resolution, but we plan to perform more experiments in the future to improve that.

2.2 Line Deletion

The deletion procedure of the detected ruling line is based on the mathematic rule [9]: three collinear points (x_i,y_i) have a determinant *D* equal to zero:

$$D = \begin{bmatrix} 1 & 1 & 1 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} \quad (1)$$

Here, the points *(x1,y1)* and *(x2,y2)* are detected from the document image while the *(x3,y3)* is calculated by posing *D=0*.

First the thickness of the detected ruling line is also detected, by examining the area carefully. As the estimation of the ruling line has been made by the upper profile, the considered possible position (see §2.1) should be the start of the ruling line in the points that it does not intersect with text. Thus, the pixel line in the detected position is scanned and the pixel columns with black pixel at this pixel line and white at the upper one are examined vertically from this point on, counting the continuous black pixels. The amount of black pixels, which is met in most of the pixel columns, is considered to be the *thickness* of the ruling line.

Having the thickness, the calculation of points (x_1,y_1) and (x_2,y_2) is the next step. It is easily done by the procedure described in §2.1, considering a vertical part of the same document image. In our case, a vertical zone wide as much as W/5 was considered, where W is the width of the page. We considered the first of the five vertical zones for the detection of point (x_1,y_1) and the last one for the detection of point (x_2,y_2). It does not really matter the width of the vertical zone or its position but due the aliasing problems that are present in the digital images, the furthest the two points, the best the accuracy in the estimation (see §4). Moreover, the vertical zones should be enough wider than the possible scan noise of the page. Remember that we have not preprocessed the image to clean it. If the width of the vertical zone is less than double of the scan noise width, the point estimation will not be correct.

Having, already, a point estimated for the left most vertical zone, as before, and another one for the right most vertical zone, these are not the (x_1,y_1) and (x_2,y_2). The pixel lines in the detected positions are scanned and the central pixel of the first pixel column with amount of black pixels equal to *thickness* in the left vertical zone is the point (x_1,y_1), while in the right one is the point (x_2,y_2).

Next, setting *x3=1* to *W*, the *y3* can be calculated for each column of the ruling line by posing *D=0* at (1). Then:

$$y_3 = \left((-y_1) * \det\begin{bmatrix} 1 & 1 \\ x_2 & x_3 \end{bmatrix} + y_2 * \det\begin{bmatrix} 1 & 1 \\ x_1 & x_3 \end{bmatrix} \right) / \det\begin{bmatrix} 1 & 1 \\ x_1 & x_2 \end{bmatrix} \quad (2)$$

The point (x_3,y_3) should be collinear to the points (x_1,y_1) and (x_2,y_2). If the (x_3,y_3) is black, the height of this black area in the pixel column is checked and if it is less or equal to *thickness+2* it is changed to white, otherwise either it is broken ruling line (if it is not black), or there is character (if larger). The value 2 is added to thickness in order to cover the aliasing problems of digital images.

After the deletion of the detected ruling line the area from the beginning of the page till the detected position of the page plus an *offset=5*thickness,* is turned to white in order to calculate upper profile for the next ruling line. The factor 5 in the offset could be anything big enough to clean the image up to after the ruling line, in order to be able to detect the next one. The whole procedure is repeated till the end of the page is reached.

Finally, vertical ruling lines can be detected and deleted by repeating the whole procedure after rotating the page 90 degrees of the page and considering as width, the height of the page. In case that there are ruling lines very close to each other, as the vertical lines of Fig.2, special care should be taken for the above mentioned *offset* not to overcome the second ruling line without detecting it.

Please note that with the above mentioned methodology, the presence of scan noise is not a problem. However, in case that it is long enough, it could be considered a line and be deleted.

3. EXPERIMENTAL RESULTS

As in our previous work [8], in order to give objective results for our proposed approach and comparative to the previous one, the evaluation methodology described in [5] was used. In that paper, synthetic data is employed to provide ground truth for each ruling line. The authors use 5 images of ruling lines and 10 images of Arabic documents, yielding 50 test images. Evaluation is via recall/precision and weighted harmonic mean F1 metrics, defined as:

$$precision = \frac{tp}{tp + fp} \quad (3)$$

$$recall = \frac{tp}{tp + fn} \quad (4)$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (5)$$

Where:

- True positive pixel (*tp*): a pixel that exists in both the detection map and the rule line ground truth map, but not in the text only map.
- False positive pixel (*fp*): a pixel that exists in both the detection map and the text only map.
- False negative pixel (*fn*): a pixel that exists in the ground truth map, but in neither the detection map or the text only map.

In our case, 10 scanned page images with ruling lines from different pads were used with 10 images of text from different languages written by different persons (3 English pages, 2 Greek, 2 German, 1 French and 2 Arabic), resulting in 100 images of text with ruling lines.

Table 1 presents comparative results for the old and the new technique, distinguished by language as we wish to check for language dependence. In order to have comparative results, the scan noise removal was used here, too, although not necessary.

Table 1. Comparative results of the new and the old techniques.

Proposed Technique	*English 30 images*	*Greek 20 images*	*German 20 images*	*French 10 images*	*Arabic 20 images*	*Total 100 images*
Precision	0.82	0.88	0.73	0.70	0.94	0.81
Recall	0.94	0.93	0.92	0.93	0.90	0.92
F1	0.88	0.90	0.81	0.79	0.91	0.86
Technique [8]						
Precision	0.81	0.88	0.68	0.65	0.96	0.76
Recall	0.93	0.93	0.90	0.93	0.90	0.91
F1	0.86	0.89	0.75	0.75	0.93	0.81

In Fig.3 an example of original image (Fig.3a) is shown, processed by the old [8] (Fig.3b) and the new (Fig.3c) techniques. Moreover, in (Fig.3d) you can see a typical error of our system. There are some cases that although the ruling lines have been detected successfully and deleted, small parts of them can escape due to the aliasing problems of the digital images that can affect the accuracy in the mathematical estimation of the position of the ruling line.

To estimate the computation time of the proposed technique it was implemented in Matlab and run 3 times for the whole database of synthetic images (100 images, 3270 lines) in a PC with Intel Core 2 Duo 3.40 GHz. The mean computational cost per image is 22.92 sec.

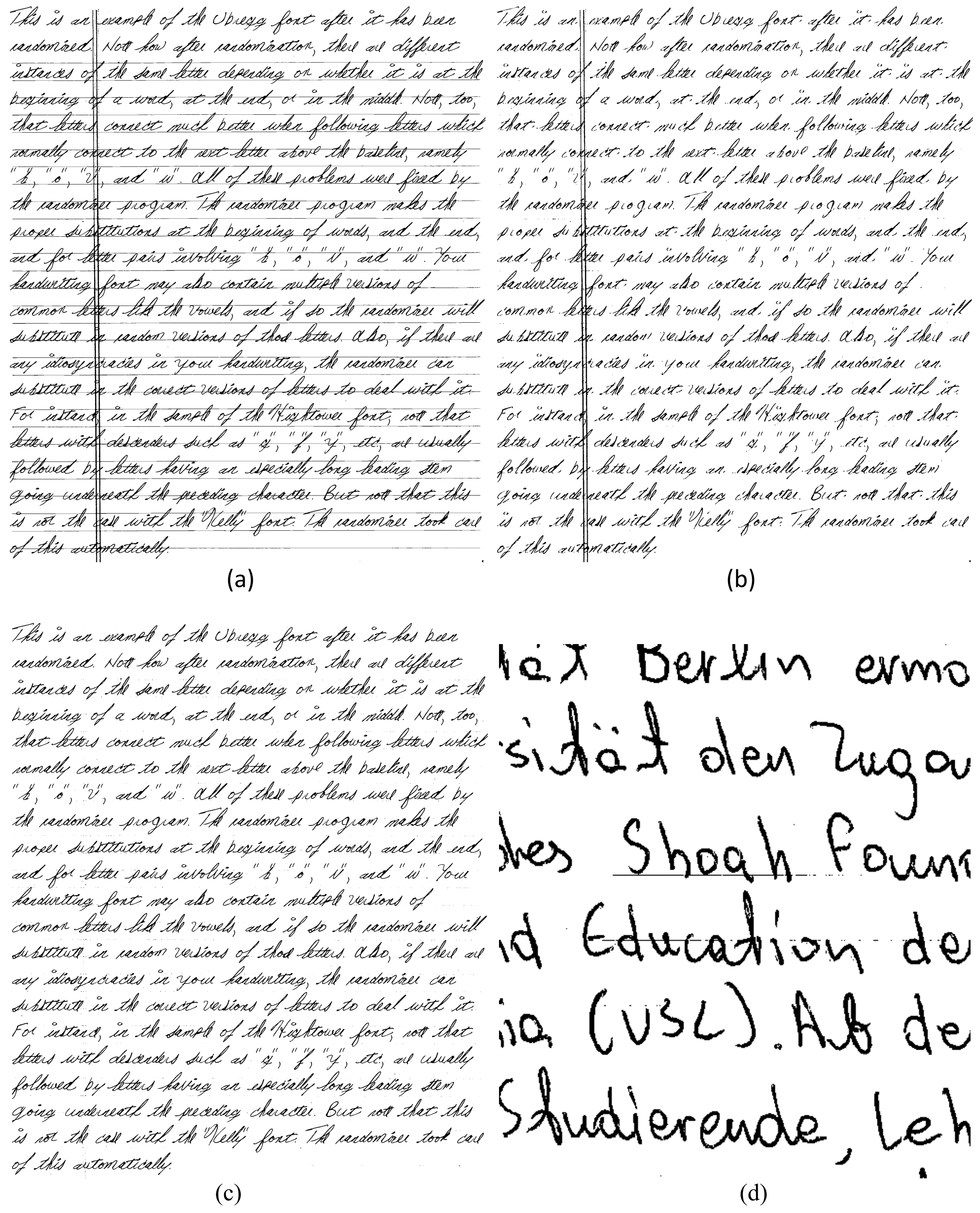

Figure 3. (a) Original Image (b) Result from Old Technique[8] (c) Result from New Technique (d) Typical error.

4. CONCLUSION

In this paper, we presented a novel technique for ruling line detection and deletion. The upper profile of the document image is used for the ruling line detection, while it is calculated again after each deletion for the next part of the document image. The deletion procedure of the detected ruling line is based on the fact that the coordinates of three collinear points have a determinant equal to zero. The whole procedure is repeated on the both dimensions of the image in order to remove horizontal and vertical ruling lines, if any.

We gave comparative results using 100 synthetic pages, formed by 10 pages of ruling lines scanned from different pads in combination with 10 text images in 5 different languages. That database was also used with our older system so we give more comparative results

The new technique gives results equally good or slightly better than our previous technique [8]. However this one does not require any preprocessing or postprocessing tasks. Moreover, it is more robust in the presence of skew in the image or ‘cloudy’ lines in the case of grayscale scanning and binarization of the document image.

The presence of aliasing in digital images seems to provoke some problems in our system that we wish to analyze further and maybe deal with, in the future.

5. ACKNOWLEDGMENTS

The work by Daniel Lopresti and Jin Chen is supported in part by a DARPA IPTO grant administered by Raytheon BBN Technologies.

REFERENCES

1. K. Tombre, S. Tabbone, L. Plissier, and B. Lamiroy. “Text/graphics separation revisited.” *Workshop on Document Analysis Systems*, pp. 200–211, 2002.
2. F. Cesarini, M. Gori, S. Marinai, and G. Soda. “Informys: A flexible invoice-like form-reader system” *IEEE Trans. Pattern Anal. Mach. Intell*., 20(7):730–745, 1998.
3. Huaigu Cao, Venu Govindaraju, "Preprocessing of Low-Quality Handwritten Documents Using Markov Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1184-1194, July, 2008.
4. P. Dosch, K. Tombre, C. Ah Soon, and G. Masini. “A complete system for the analysis of architectural drawings” *IJDAR*, 3(2):102–116, 2000.
5. Wael Abd-Almageed, Jayant Kumar, David Doermann, "Page Rule-Line Removal Using Linear Subspaces in Monochromatic Handwritten Arabic Documents," *10th International Conference on Document Analysis and Recognition*, pp. 768-772, 2009.
6. K.R. Arvind, J. Kumar and A.G. Ramakrishnan, “Line Removal and Restoration of Handwritten Strokes,” *International Conference on Computational Intelligence and Multimedia Applications*, vol. 3, pp. 208-214, 2007.
7. H. Cao, R. Prasad, and P. Natarajan. “A stroke regeneration method for cleaning rule-lines in handwritten document images” *In Proc. of the MOCR workshop at the 10th international Conference on Document Analysis and Recognition*, 2007.
8. Lopresti, D. and E. Kavallieratou, “Ruling Line Removal in Handwritten Page Images”, *IEEE Proc. of 20th International Conference of Pattern Recognition (ICPR 2010)*, pp. 2704-2707, 2010.
9. Kenneth Eriksson, Donald J. Estep, Claes Johnson, Applied Mathematics, Body and Soul: Calculus in several dimensions, Springer, 2004.