Skew Correction & Rule Line Removal in Devanagari

[1]Mahesh Jangid, [2]Dr. Sumit Srivastava
[1,2]Dept. of CSE, Manipal University Jaipur, Rajasthan, India

Abstract
In this paper, we present a technique that first align the page (Skew Correction) after that removing the pre-printed rule lines in handwritten Devanagari document images. This paper is first approach toward the rule line removal from Devanagari Script. In survey we found a lot of work has done on English, Greek, Arabic etc. language which has different script and style to write. We use common ruling line properties such as uniform width, predictable spacing, position vs. text etc. to remove the rule line. Dataset is prepared by 12 different writers who used the rule-line page and handwritten pre-printed rule-line documents are scanned on 200dpi. The average values of precision, recall and harmonic mean are obtained 93.24 %, 95.36 % and 94.27 % respectively.

Keywords
Document Image, Ruling Line Detection, Ruling line Removal

I. Introduction

Optical character recognition system is a field where we are finding the handwritten and printed characters and requires the preprocessing stage that enhances the quality of the document image by removing noise elements one of the noise element is pre-printed rule line which is required to remove before going to recognition phase. Line processing is a necessary task in form and invoice processing, engineering drawings etc. Rule lines are used in handwritten documents as a guide to make it easier to write neatly but handwritten text overlapping with the rule-lines poses serious problems for their recognitions. Rule-line detection and removal are necessary tasks in any handwritten document recognition. Rule line generally hassome common features [1] (1) Rule-lines are uniform in thickness (2) Rule-line's position is predictable on the page (3) Rule-line have lighter in color and thickness than the handwritten text on page (4) Handwritten text overlap the rule line. The binarisation process makes the uniform thickness of rule line into variable in thickness even in a same page document image and broken rule-lines shown in figure 1.

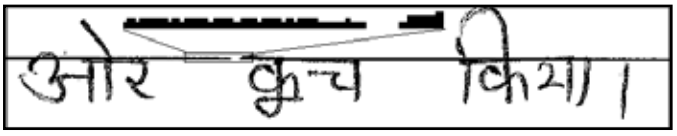

Fig. 1: Variable Thickness and Broken Rule-Line

Rule-line removal approaches has proposed by many researchers for English, Arabic etc. language but not for Devanagari script or Hindi language which is an oldest one that is used to write many languages such as Hindi, Nepali, Marathi, Sindhi and Sanskrit where Hindi is the third most popular language in the world and it is the national language of the India [2]. 300 million people use the Devanagari Script for documentation in central and northern parts of India. Devanagarihas different way to write on rule-lined page shown in fig. 2. Fig. 2 shows that Devanagari Script writes below the rule line and write a header line on the top of a word while English and Arabic languages is written on the rule line and there is no header line. When rule-line is removed from English and Arabic it's completely removed but in case of hind or Devanagari a part of rule-line as a header line of a word is not removed only other part is removed.

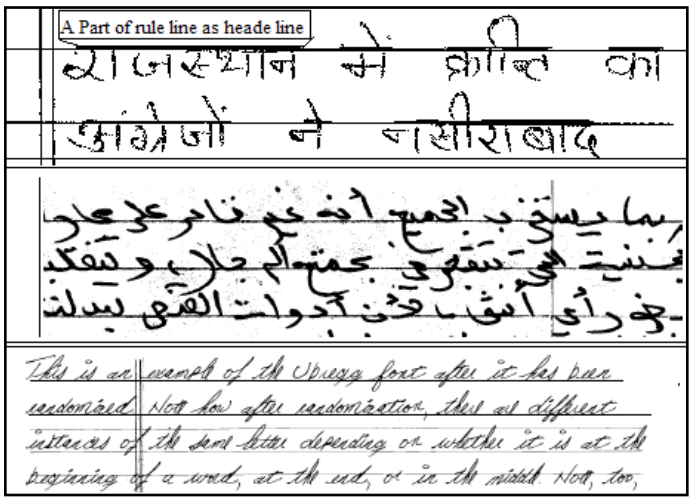

Fig. 2: Hindi, Arabic and English Language

From the literature survey [3], Existing approaches for rule-line removal can be classified as heuristic-based or model-based. Rule-lines are detected and removed using the projection profiles, Hough-Transform, Run-lengths or Morphological operators. A simplest way to detect rule line is Projection Profile in which a horizontal histogram is used to locate the center point of the horizontal rule-lines [3, 7]. Projection Profile based methods are very sensitive to the skew of the document image and measurement of the rule-line thickness is also difficult. Cao et al. [7] proposed a new method based on partitioning the document image into vertical zones and projection profiles were computed for each zone but fail to find suitable width for zone. To detect the rule line, Hough transforms based methods also used but they are computationally very slow. Dilation and erosion operations [8] (Morphological based methods) use a structuring element to remove rule-lines but to find structuring element is difficult due to the large variation in rule-line thickness.

This paper presents a technique first to skew correction for the document image by the help of upper profiling and detect the skew angle after that detection of rule-line by tracing the pixel by pixel horizontally then according to the detected rule-line classify the rule line and header line. Dataset is prepared by the help of 12 different writers. All the handwritten pre-printed rule line documents are scanned on 200 DPI.

The section II, described the proposed system while experimental results and a discussion of future work are presented in section III and IV respectively.

II. System Description

A. Skew Correction

To locate the rule-line in the document image can be by horizontal histogram but this method is very sensitive with skew of the document image and estimation of thickness of rule-line is also difficult. Firstly skew correction must be done before going to detect and removal the rule line.

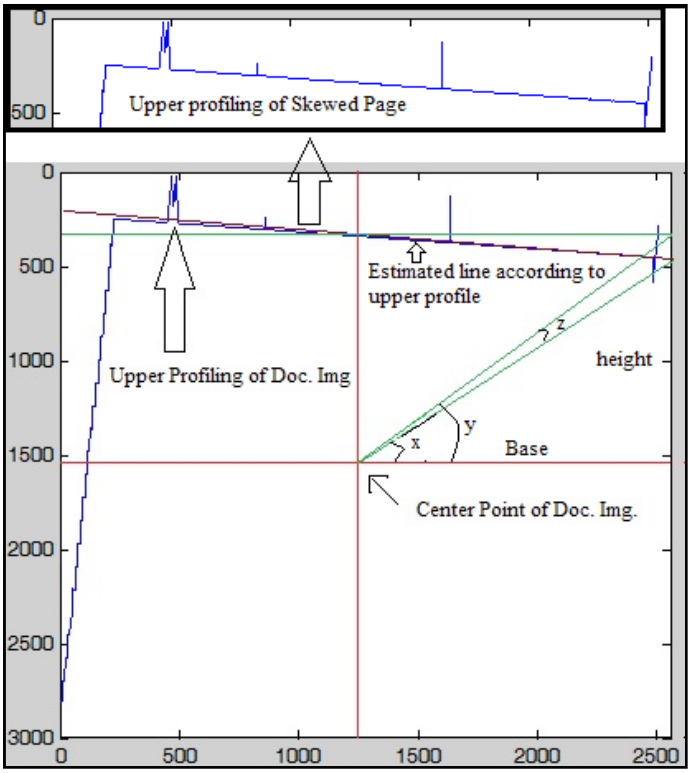

Fig. 3: Estimating the Skew Angle by Upper Profiling

Fig. 3 shows the upper profilewhich estimated by find the first foreground pixel in each column of the document image, starting from top. Due to the noise or broken rule line, upper profile has some peaks. The adjacent pixels of the rule-line (or upper profile) that are included in the page would differ vertically by 0 when there is no skewed page or a little skewed page and 1 when page is skewed. In case that there are no rule-line is present 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. After finding the top rule-line, a straight line is found that is used to determine the slope or skew angle z. Skewed page is divided into four parts and a parallel line is also drawn where the rule-line cross the vertical line. These two lines are created two triangle otherwise they create some triangle. Equation 1 is used to find the angle x, y and z by subtracting y from x. The angle value of the z may be positive or negative depends on the page is skewed clockwise or anticlockwise.

$$\theta = \tan^{-1} (\text{height}/ \text{base}) \quad (1)$$

B. Line Detection

Skew corrected page have rule-line now next phase is to detect the location of the rule-line. As discussed above horizontal histogram may be used to detect the rule-line but still may be a chance that document page is still have 1 or 2 degree skew at that time horizontal histogram fails to detect the rule-line. So to detect the rule line a new technique is proposed that is less sensitive with little skewed page.

Some preprocessing steps are applied before going to detect the rule-line (1) Median filter [2,2] is applied that is a nonlinear operation often used in image processing to reduce "salt and pepper" noise. A median filter is more effective than convolution when the goal is to simultaneously reduce noise and preserve edges. (2) After median filter, an imdilate function is applied with structuringdiamond 1 element to make the rule-line little bit uniform.

This Line detection method starts with searching a black pixel (foreground) from top left corner. If a black pixelim(x, y)(x represents row and y represents column) is met then next back pixel is searched. There are three places where the black pixel can meet at location im(x-1, y+1), im(x, y+1) and im(x+1,y+1). Second pixel is given the higher priority than other and checks that it is foreground (black) pixel or not. If it is a foreground pixel then same process is repeated for next three pixels. If second is not a foreground pixel then priority is given to first and three depends on the skew angle it's clockwise or anticlockwise. If foreground pixels are met continuously they saved as rule line shown in fig. 4.

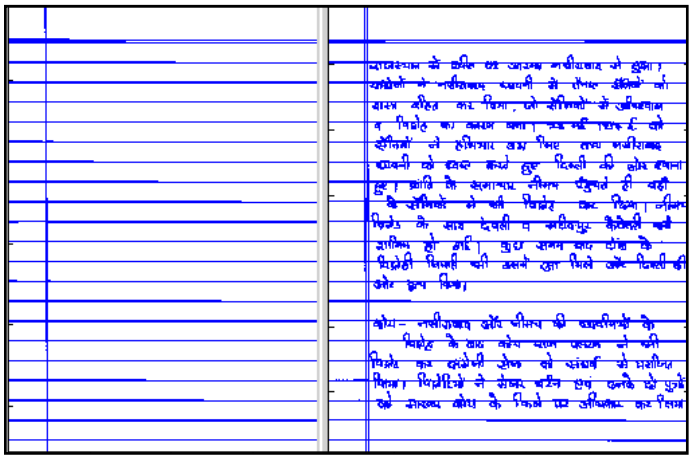

Fig. 4: Rule-Line Detected from Document Image

C. Rule-line Removal

Fig. 4 shows the rule-line exists in the document image. The next step is to remove the rule-line from the document image. This task is little bit typical due to the variable thickness of the rule-line and existence of header line or not. So the thickness of the detected rule-line is estimated first, by examining the area carefully. In the skew correction section II (A) the estimation of the rule-line has been made by the upper profile, the considered possible position should be the start of the rule-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 the 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 amounts of black pixels are varied due to variable thickness of rule-line. So average is taken to find the thickness of rule-line.

In Devanagari Script, every word has a header line that is made after written the characters. This header line is used to separate the words. The thickness of the header line is greater than the rule-line generally. But something, due to the presents of rule-line, writers don't make the header line. In this case, rule-lines exist on the word are considered as a header line, which should not be removed from the documents image. Algorithm determines that what portion of the rule-line only image (estimated in section II (b)) is an actually rule-line and what portion is a header line. If both the rule-line only image and document image have same black pixels then there is a chance that it may be a part of rule line or header line. First the thickness is calculated and also looks the next 30 by 10 pixels region there are white pixels. If the estimated thickness is less the actual rule-line thickness and estimated region also have white pixels means it's a rule-line portion, delete the rule-line and if the thickness is less than actual rule-line and estimated region have black (foreground) pixels means it's a header line. The same procedure is used to remove the vertical rule-line after rotating the image by 90 degree. Text only image of figure 4 is shown in

fig. 5, after removing rule-lines.

Algorithm:-Rule-line Removal
Step 1:- imr (h,w) rule-line only, im(h,w) document image
Step2: Estimation the thickness of rule line
Step 2: loop i=1 to height
Step 3 loop j=1 to width
Step 4 if (imr (i,j) and im (i,j) have black pixel)
Step 5 If ((Count the black pixels < =thickness) and im (i+10: i+30, j: j+10) = white pixels)
Delete rule-line
Step 6 exit

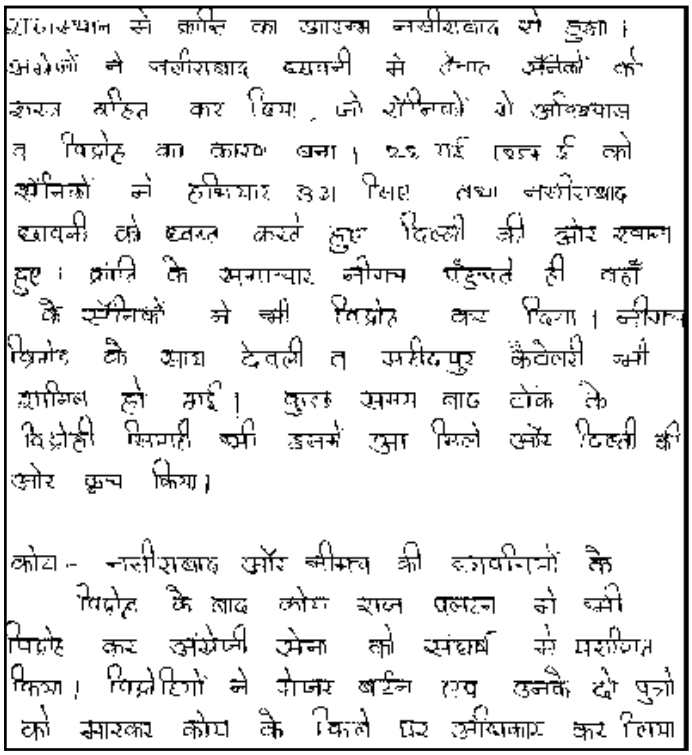

Fig. 5: Text Only Image (Final Result)

III. Experimental Results

The way by which result is estimated by W. Abd-Almageed et al. [4] is used, to estimate the experiment results of our work. 12 document images are used to done experiment. To estimate the result, rule-line is removed manually from the document image by the help of Microsoft® paint tool.

We compute recall, precision values and harmonic mean (F1 score) to evaluate our method. If a rule-line pixel detected by our method is also a rule-line pixel then it is counted as true positive (Tp) and if a rule-line pixel detected by our method is not a rule-line pixel then it is counted as false positive (Fp). Finally those rule line pixels which are missed by our method, are counted as false negative (Fn). Using these values, we compute precision, recall and F1 score as follows.

$$\Pr ecision = \frac{Tp}{Tp + Fp}$$

$$\mathrm{Re}call = \frac{Tp}{Tp + Fn}$$

$$F_1 score = \frac{2 * \Pr ecision * \mathrm{Re}call}{\Pr ecision + \mathrm{Re}call}$$

Table 1 shows the experiment results. The average values of precision, recall and harmonic mean are 93.24, 95.36 and 94.27 respectively. The average image size is 2300 by 2800 means 644000 pixels after binarization process. To estimate the computation time of the proposed system it was implemented in Matlabin a laptop with Intel® Core™ i3-2310 CPU @ 2.0Ghz 2.10 Ghz and 4 GB RAM. The mean computational time per image is 25 seconds.

Table 1: Show the Experiment Results

Dataset	Experiment Results		
	Precision %	Recall %	F1score %
Image 1	95.39	96.52	95.96
Image 2	91.22	95.00	93.08
Image 3	93.04	95.33	94.17
Image 4	91.01	91.46	91.23
Image 5	92.73	95.81	94.25
Image 6	95.36	92.49	93.90
Image 7	91.51	95.90	93.65
Image 8	91.48	95.85	93.61
Image 9	93.11	95.44	94.26
Image 10	95.11	95.42	95.26
Image 11	91.86	98.06	94.85
Image 12	97.05	97.00	97.02
Average	93.24	95.36	94.27

Fig. 6 shows the typical errors found after the experiment. As every script has character and numeral. Due to the confusion our method has a header line on numeral shown in fig. 6(a). In Devanagari script when we write words there may be a case that there is a big gap between two characters of same word. So our method treats two separate words instead of one word shown in fig. 6(b). As we already assumed that rule-line exist on a word, is treat as a header line due to this assumption there is a two header-line on a word shown in fig. 6(c).

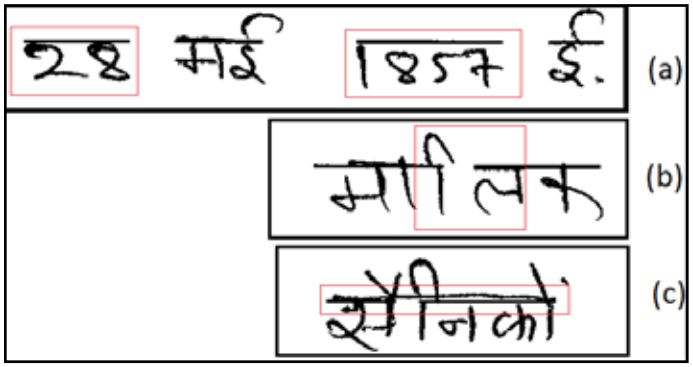

Fig. 6: Typical Error

IV. Future Work

As Table 1 shows the good results to removal of rule-line in Devanagari script but still there is a chance to improve the recognition accuracy. Fig. 6 shows the some typical error found in this work so my future work is to overcome all this issue in future. Moreover, we plan to extend the work to detecting and removing rule line in any direction, rather than horizontal and vertical lines, which frequently occur in photo-copied handwritten document. Datasets are available for download at [9] free of cost.

References

[1] Lopresti, D., E. Kavallieratou,"Ruling Line Removal in Handwritten Page Images", IEEE Proc. of 20th International Conference of Pattern Recognition (ICPR 2010), pp. 2704-2707, 2010.

[2] Mahesh Jangid, Renu Dhir, Rajneesh Rani, Kartar Singh Siddharth,"SVM Classifier for Recognition of Handwritten Devanagari Numeral", International Conference on Image Information Processing (ICIIP-2011), pp. 1-5, 3-5 Nov. 2011

[3] K. R. Arvind, J. Kumar, A. G. Ramakrishnan,"Line removal and Restoration of Handwritten strokes", Intl. Conf. on Comp. Intelligence and Multimedia Applications, IEEE CS Press Vol. 3, pp. 208–214, 2007

[4] W. Abd-Almageed, J. Kumar, D. Doermann,"Page Rule-Line Removal using Linear Subspaces in Monochromatic Handwritten Arabic Documents", Intl. Conf. on Document Analysis and Recognition, pp. 768–772, 2009.

[5] J. Kumar, W. Abd-Almageed, L. Kang, D. Doermann, "Handwritten Arabic Text Line Segmentation using Affinity Propagation", Document Analysis Systems, pp. 135–142, 2010.

[6] A. K. Chhabra, V. Misra, J. F. Arias.,"Detection of horizontal lines in noisy run length encoded images: The fast method", Intl. Work. on Graphics Recognition, Methods and Applications, pp. 35-48, 1996.

[7] H. Cao, R. Prasad, P. Natarajan,"A stroke regeneration method for cleaning rule-lines in handwritten document images", MOCR, pp. 1-10, 2009.

[8] J. Said, M. Cheriet, C. Suen,"Dynamical morphological processing: A fast method for base line extraction, Intl. Conf. on Document Analysis and Recognition, pp. 8-12, 1996.

[9] [Online] Available: http://www.maheshjangid.wordpress.com/downloads/