Sādhanā Vol. 30, Part 1, February 2005, pp. 69–85. © Printed in India

A novel technique for estimation of skew in binary text document images based on linear regression analysis

P SHIVAKUMARA*, G HEMANTHA KUMAR, D S GURU and P NAGABHUSHAN

Department of Studies in Computer Science, Manasagangotri, University of Mysore, Mysore 570 006, India
*e-mail: hudempsk@yahoo.com

MS received 9 September 2003; revised 16 January 2004

Abstract. When a document is scanned either mechanically or manually for digitization, it often suffers from some degree of skew or tilt. Skew-angle detection plays an important role in the field of document analysis systems and OCR in achieving the expected accuracy. In this paper, we consider skew estimation of Roman script. The method uses the boundary growing approach to extract the lowermost and uppermost coordinates of pixels of characters of text lines present in the document, which can be subjected to linear regression analysis (LRA) to determine the skew angle of a skewed document. Further, the proposed technique works fine for scaled text binary documents also. The technique works based on the assumption that the space between the text lines is greater than the space between the words and characters. Finally, in order to evaluate the performance of the proposed methodology we compare the experimental results with those of well-known existing methods.

Keywords. Document analysis; OCR; connected component; boundary growing; scaled documents; linear regression analysis; skew detection.

1. Introduction

Conversion of a paper document to electronic format is routinely performed for various purposes. This includes archiving and many other applications. The principal stages in a document conversion system are scanning, binarization, region segmentation, text recognition and document analysis, which includes steps from simple spelling correction to natural language understanding. During the scanning process, the document may not be fed properly into the scanner. The text lines in the document images would thus not be horizontal and may cause problem in segmenting the document image to extract its layout structure. For example, in the most commonly used method of profiling, in the presence of skew there would not be valleys in the projection and segmentation would not be possible. Skew detection and removal is, thus, a very important stage in document analysis (Amin & Fischer 2003).

Most optical character recognition (OCR) systems are very sensitive to skew in text document images. The methods of OCR systems may fail to obtain the expected results due to problems in accurately detecting skew angle and removing noise. Even a small degree of skew existing in a given text document image for recognition, results in failure of segmentation of complete characters from words in it since the space between the characters, words and text lines is reduced. Similarly, document image mosaicing methods fail to obtain a mosaiced image from its split images in presence of skew in one of the split images. This is because the methods work based on the Pattern Matching Approach (PMA). PMA is obtained by generating the strings of column sums (SCS) of the split images. SCS is the string of the sum of values of pixels present in each column of the split image. When both of the split images are skewed by different angles there is no match in SCS for these split images. Therefore, mosaicing becomes impossible (Shivakumara *et al* 2001). Hence, accuracy in finding a skew angle is also much important in the field of DIM and OCR rather than estimating skew detection for the skewed document.

There are several methods for skew detection. These methods are based on projection profile, Hough transform, Fourier transform, nearest Neighbour clustering, and interline cross correlation.

In the projection profile method, a series of projection profiles are obtained at a number of angles close to the expected orientation and the variation is calculated for each of the profiles. The profile which gives maximum variation corresponds to the projection with the best alignment to the text lines and that projection angle is the actual skew angle of the document. Baird (1987) proposes this method and states that the skew angle should be limited to $\pm 15^\circ$ to achieve high accuracy. However, the accuracy depends upon the angular resolution of the projection profile. However, the method is time-consuming and its accuracy reduces when the documents are noisy and containing character fragments.

Srihari & Govindaraju (1989), Hinds *et al* (1990) and Pal & Chaudhuri (1996) have proposed skew detection techniques based on the Hough transform (HT). The HT maps each point in the original (x, y) plane to all points with (ρ, θ) Hough space of lines through (x, y) with slope θ and distance ρ from the origin where $\rho = x \operatorname{Cos}\theta + y \operatorname{Sin}\theta$ for $0 \leq \theta < \prod$. The peak in the Hough space represents the dominant line and it is skew. The major draw back of this method is that it is computationally expensive and is difficult to choose a peak in the Hough space when text becomes sparse.

Postl (1986) proposed a method based on Fourier transform (FT). In this method the direction for which the density of the Fourier space is the largest, gives the skew angle. It is found that this Fourier method is computationally expensive for large images.

A bottom-up technique for skew estimation based on nearest neighbour clustering (NNC) is described by Hashizume *et al* (1986). In this work, nearest neighbours of all connected components are found, the direction vector for all nearest neighbour pairs are accumulated in a histogram and the histogram peak is found to obtain a skew angle. Since only one nearest neighbour connectivity is made for each component, connection with noisy sub parts of characters would reduce the accuracy of the method.

Yan (1993) introduced a method for detecting the skew angle of an image using cross-correlation between the text lines at a fixed distance. It is based on the observation that the correlation between vertical lines in an image is maximized for a skewed document, in general, if one line is shifted relatively to the other lines such that the character base line levels for two lines are coincident. It is found that the proposed method is computationally expensive as well as it being less accurate.

Gatos *et al* (1997) have proposed a new skew detection method based on the information existing on a set of equidistant vertical lines. Further, the method could consider all the image pixels that lie on a set of equidistant vertical lines. By using these pixels they construct a correlation matrix between vertical lines. Gatos *et al* (1997) did not relate every pixel of a line to all pixels of other lines but only to those pixels that lie in a specific region defined by the expected maximum skew. However the method works only for small skewed documents.

Liolios *et al* (2002) have proposed a generic skew detection method for any type of preprinted form which is based on power spectral density (PSD) of the form horizontal projection profile. However, the method is accurate for only small amounts of skew of documents.

The same authors (Shivakumara *et al* 2002, 2003) have proposed skew detection techniques for estimation of skew angle of a text binary documents based on linear regression analysis. The techniques are very simple compared to the above methods since the methods do not involve any expensive methods like Hough transform and others to determine skew angles for documents. The techniques work based on segmentation text lines from skewed documents. However, the skew angle should be limited to $\pm 10^\circ$ to achieve accuracy since the methods fail to segment complete text line from the skewed document as skew angle increases.

A review of the literature reveals that the methods give accuracy but they are computationally expensive. If the methods are inexpensive they offer less accuracy. In addition to this, the existing methods have not considered the problem of skew detection in scaled text documents. The problem of scaling is obvious during scanning the document. Hence, there is a need for research to develop a method for providing high accuracy at low costs.

In this paper, a simple and efficient method in terms of accuracy and computations, to estimate the skew angle of a scanned document image is presented. The proposed method uses the boundary growing approach to extract the lower and the uppermost coordinates of pixels of characters of text lines present in the document. In order to estimate skew angle for a document the method substitute the lower and upper most coordinates of pixels of characters in linear regression analysis (LRA). The proposed method is also used to determine skew angle for scaled documents. However, the method assumes that the space between the text lines is greater than the space between the words and characters.

The organization of the paper is as follows. Section 2 presents a technique for estimation of skew angle for skewed text document. In § 3 we have shown that the proposed methodology is works for scaled documents also. The experimentation and comparative study is reported in § 4. Finally, results and discussion are presented in § 5.

2. Proposed work

This section presents an algorithm to detect the skew angle of the given skewed text document image based on the study of the direction of the text lines in the image. The method also considers scaled documents to estimate skew angle. The direction of the skewed text lines is obtained based on the boundary-growing approach (Duda & Hart 1973; Gonzalez & Woods 1997). The boundary-growing method helps in extracting the lower and uppermost coordinates of pixels of characters of text lines, which are then subjected to the following linear regression analysis formula to estimate the skew angle of the document with respect to the horizontal axis. In addition to this, skew angle estimation by rectangle fitting is also presented. The proposed work assumes that the space between the text lines is greater than the space between the words and characters.

The regression line formula is a statistical method for finding the equation of best fit given a set of points. We determine the equation of best fit for a line $Y = A + B \times C$,

$$B = \frac{n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{n\sum x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}, \qquad A = \frac{\sum_{i=1}^{n} y_i - B\sum_{i=1}^{n} x_i}{n}, \tag{1}$$

where B is the value of the slope, x and y are the coordinates of the pixels and n is the number of coordinates.

The skew angle is computed using the following formula

$$- \quad -\theta = \tan^{-1}(B). \tag{2}$$

The proposed methodology is divided into 4 subsections. The boundary-growing approach is described in the first subsection. The second subsection explains the procedure for estimating a skew angle using the lowermost coordinates of the pixels of characters. The third subsection describes the procedure to estimate skew angle using the uppermost coordinates of the pixels of characters, while the fourth subsection presents the averaging method by combining the two above.

2.1 *Boundary-growing approach to extract lower- and uppermost pixels of characters*

This subsection presents the boundary-growing approach to extract the coordinates of the lowermost and uppermost pixels. The method finds the lower and the uppermost coordinates to fix the boundary for characters present in the text line and then the method allows the boundary of character to grow until it reaches a pixel of neighbour character to get lower and upper most coordinates of neighbour character. This procedure is repeated till the end of the text line. This is shown in figure 1. This is possible only when there is more space between the text lines compared to the space between the words and characters. Skew angles are computed separately using lowermost and uppermost coordinates by substituting the coordinates into the LRA formula equations (1) and (2). This is repeated for two to three text lines in the document. The average of all the skew angles of text lines gives the actual skew angle of the entire document.

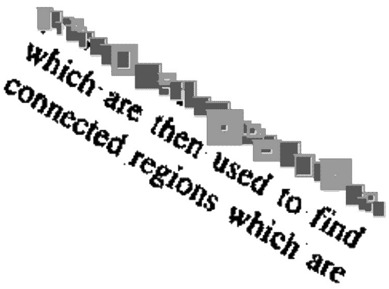


Figure 1. Boundary growing approach for 30 degree skewed document.

```
Algorithm: Boundary growing approach (BGA)
Input: Skewed text document
Output: Direction of the text line
Method:
    Step 1. Identify the first black pixel of character of first skewed text line in the document
    Step 2. Fix the boundary for the character using $X$min, $X$max, $Y$min and $Y$max coordinates of the character.
    Step 3. Compute the centroid for the rectangle using $X$ and $Y$ coordinates of rectangle $C[x] = X_1 + X_2/2$ and $C[Y] = Y_1 + Y_2/2$.
    Step 4. Draw one more rectangle by incrementing one unit length of rectangle from its centroid.
    Step 5. Allow this boundary till it reaches next pixel of neighbour character by repeating step 4.
    Step 6. If the number of boundaries exceeds the threshold ($T$) then go to next line.
    Step 7. Repeat the above procedure for two or three text lines for estimating the skew angle for the entire document.
Method end
BGA end
```

2.2 *Skew angle estimation using lowermost coordinates of pixels of characters*

This section presents a procedure to estimate skew angle by considering the lowermost coordinates of pixels of characters of text lines present in the document. The lowermost coordinates of pixels of characters are obtained using the above boundary growing approach (§ 2·1). The coordinates are subjected to linear regression analysis formula (1) to obtain skew angle, (2) for each text line in the document image. The procedure is repeated for two to three text lines in the document. Finally, the average of skew angles of text lines gives the actual skew angle of the entire document. Let $L\theta$ be the skew angle of each text line in the document. Let N be the number of text lines. Then the average of the θ can be computed using the following procedure. Hence $Al\theta$ is the actual skew angle for entire document.

$$Al\theta = \left(\sum_{i=1}^{N} L\theta_i\right) \Big/ N. \tag{3}$$

```
Algorithm: Skew estimation using lower most pixels (SELP)
Input: Skewed text document
Output: Skew angle determination
Method:
    Step 1. Employ the BGA for skewed text document (section 2·1)
    Step 2. Extract only lower most coordinates of rectangle of each character present in the text line
    Step 3. Compute the slope value of text line by substituting the above coordinates to the equation 1
    Step 4. Skew angle is determined by using the equation 2.
    Step 5. Repeat the above steps for two or three text lines
    Step 6. Actual skew angle is estimated by averaging the two or three skew text lines skew angles (refer to (3)).
Method end
SELP end
```

2.3 *Skew angle estimation using upper most coordinates of pixels of characters*

This section presents a procedure to estimate skew angle for skewed documents using the uppermost coordinates of pixels of characters of text lines present in the document image. The upper most coordinates of pixels are obtained by using the boundary-growing approach (§ 2·1). These coordinates are subjected to linear regression analysis formula (1) to compute a skew angle, (2), for the skewed document image. This is repeated for two to three text lines in the document. Finally, the average of the skew angle gives the actual skew angle for entire text document. Let $u\theta$ be the skew angle of each text line and N be the number of text lines then the average $Au\theta$ can be computed using the following formula is as follows

$$Au\theta = \left(\sum_{i=1}^{N} u\theta_i\right) \Big/ N. \tag{4}$$

```
Algorithm: Skew estimation using uppermost pixels (SEUP)
Input: Skewed text document
Output: Skew angle determination
Method:
    Step 1. Employ the BGA for skewed text document (section 2·1)
    Step 2. Extract only uppermost coordinates of rectangle of each character present in the text line
    Step 3. Compute the slope value of text line by substituting the above coordinates to the equation 1
    Step 4. Skew angle is determined by using the equation 2.
    Step 5. Repeat the above steps for two or three text lines
    Step 6. Actual skew angle is estimated by averaging the two or three skew text lines skew angles (refer the equation 4).
Method end
SEUP end
```

2.4 *The average method to obtain better approximation*

This section describes a procedure to compute the average angle by considering the above two angles ($AL\theta$ and $Au\theta$) of a skewed document. This is essential because in general the distribution of pixels of text lines is divided into three zones, namely upper, lower and middle. For the lower case letters in common written English, the frequencies of occurrences of letters in the top, middle and bottom rows are 26·5%, 67% and 6·25% respectively (Caprari 2000). This indicates that the characters touching the upper zone are more in number than characters in the lower zone. Obviously, the method for skew angle using uppermost coordinates of pixels of characters gives less accuracy than lowermost coordinates of pixels. And also if the ascenders are at one side and descenders are at another side in the same text line then both the methods give less accuracy. Therefore, we propose the averaging method to overcome the drawbacks of the above two methods. The performance of averaging method and other two is presented in section 4.

```
Algorithm: Skew estimation by averaging (SEA)
Input: Skewed text document
Output: Skew angle determination
Method:
        Step 1. Employ the BGA for skewed text document (section 2·1)
        Step 2. Employ the SELP to determine skew angle (section 2·2)
        Step 3. Employ the SEUP to determine skew angle (section 2·3)
        Step 4. Actual skew angle is estimated by Averaging of skew angles estimated by
                SELP and SEUP
Method end
SEA end
```

2.5 *Skew angle detection by fitting rectangle for text lines*

This section presents a new technique to detect a skew angle of a scanned document image based on boundary-growing method. The boundary-growing method is used to fix the boundary for connected components and to find the direction of the text lines present in the text document by allowing the boundary of connected components to grow until it reaches a pixel of a neighbouring connected component. This is possible because of the fact that the space between the text lines is always greater than the space between the words and characters in printed standard documents. The procedure is repeated till end of the text line. A rectangle is drawn for the whole text line, with the help of coordinates of rectangles of the first character and the last character of the text line as shown in figure 2. The mid-points of the right and the left faces of this rectangle are joined by a line and the slope its decides the direction of that particular text line with respect to the horizontal axis.

For an instance, let (x_1, y_1) be the coordinates of the top left position of the rectangle, (x_2, y_2) be the coordinates of bottom left of the rectangle, (x_3, y_3) be coordinates of top right of the rectangle and (x_4, y_4) be the coordinates of the bottom right of the rectangle as depicted in figure 3.

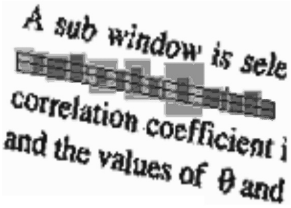


Figure 2. Rectangle is drawn for second text line using the above procedure and its middle line (slope) is also found.

Mid-points of the left face and the right face of the rectangle shown in figure 2 are calculated as follows,

$$M_1 = (X_1,\ Y_1) = ((x_1 + x_2)/2, (y_1 + y_2)/2)\,,$$
$$M_2 = (X_2, Y_2) = ((x_3 + x_4)/2, (y_3 + y_4)/2)\,. \tag{5}$$

Further, using M$_1$ (X$_1$, Y$_1$) and M$_2$ (X$_2$, Y$_2$) the slope is computed by the formula

$$\theta = \tan^{-1}((Y_2 - Y_1)/(X_2 - X_1))\,, \tag{6}$$

and the value of θ gives the direction of the text line being considered with respect to the horizontal axis. The above-described procedure is repeated for two/three text lines of the document to compute their respective θs. The average of these computed θs approximates the skew angle of the entire document.

Algorithm: Skew estimation by fixing rectangle to the text line (SERT)

Input: Skewed text document

Output: Skew angle determination

Method:

```
Step 1. Employ the BGA (§ 2·1)
Step 2. Draw rectangle for whole text line using coordinates of rectangles of first and last characters of the text line.
Step 3. Compute the mid-points using the coordinates of rectangle of whole text line (using (5)).
Step 4. Estimate skew angle using the coordinates of mid-points of rectangle of text line using the equation (using (6)).
```

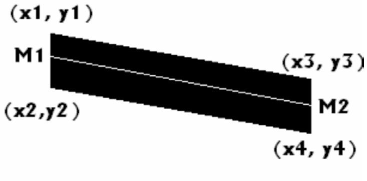


Figure 3. Mid-points of rectangle are estimated using (5).

Step 5. Repeat the above procedure for two or three text lines.

Step 6. Actual skew angle is average of two or three text lines

Method end

SERT end

3. Skew angle detection in scaled documents

This section shows that the proposed technique works well even for scaled documents. The scaling is obvious during scanning of the documents, apart from the skew of the document. The literature reveals that no methods are considered for scaled documents or to determine skew angles for scaled and skewed documents. The same method is extended to scaled and skewed documents. The algorithm is employed for different dpi as shown in figures 4–8 depicting a 30° skewed document. A comparative study of skew angles estimated for scaled and normal documents is given in § 4 (tables 5 and 6)

4. Comparative study and experimentation

For an experiments we have considered more than a hundered text document images from different books, magazines and journals. Out of them, a few are considered for processing. The documents are tested by a predefined angle varying between 0 and 30 degrees. This angle is considered the true skew angle. Here we have tested the proposed algorithm with different images for each angle, namely 3, 5, 10, 20 and 30 degrees.

Table 1 contains skew angles obtained by lowermost, uppermost and averaging methods with respect to 3, 5, 10, 20 and 30 degree predefined skew angles. From table 1 and figure 9

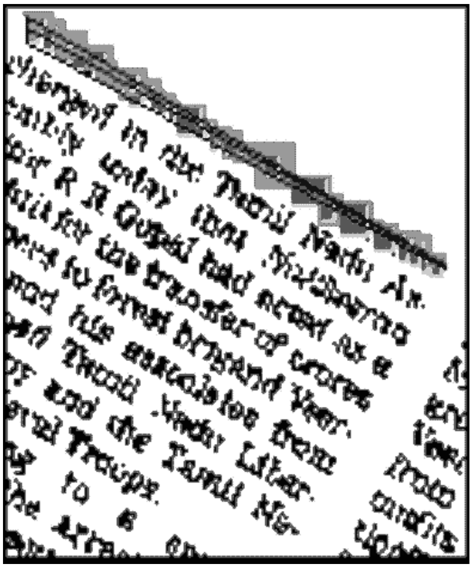
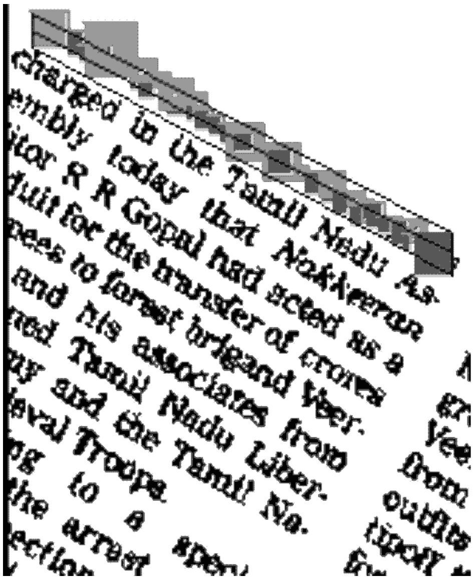


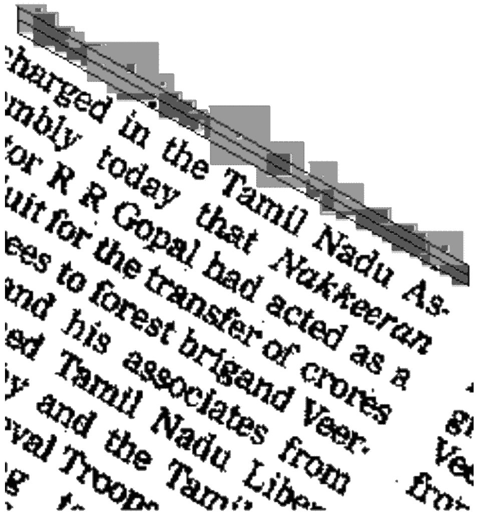
Figure 4. Document scanned at 75 dpi.



charged in the Tamil Nadu As-
embly today that Nakkeeran
itor R R Gopal had acted as a
duit for the transfer of crores
pees to forest brigand Veer-
and his associates from
ted Tamil Nadu Liber-
ay and the Tamil Na-
eval Troops
ng to a
the arrest

Figure 5. Document scanned at 100 dpi.

it is clear that the averaging method gives consistent results as compared to the other two methods since the lowermost and uppermost methods differ in determining skew angles because of distribution of pixels in the text lines. We also notice that the lowermost method gives better results compared to the uppermost pixels method. Similar conclusions can be drawn by seeing figure 9 showing the means of skew angles and predefined skew angles. From table 2 and figure 10 we observe that the averaging method gives more consistent results than the lowermost and uppermost methods (see figure 10). Standard deviations of lowermost and uppermost are changed drastically as depicted in figure 10. The rectangle-fitting method



charged in the Tamil Nadu As-
mbly today that Nakkeeran
tor R R Gopal had acted as a
uit for the transfer of crores
ees to forest brigand Veer-
nd his associates from
ed Tamil Nadu Liber-
y and the Tam
val Troop

Figure 6. Document scanned at 150 dpi.

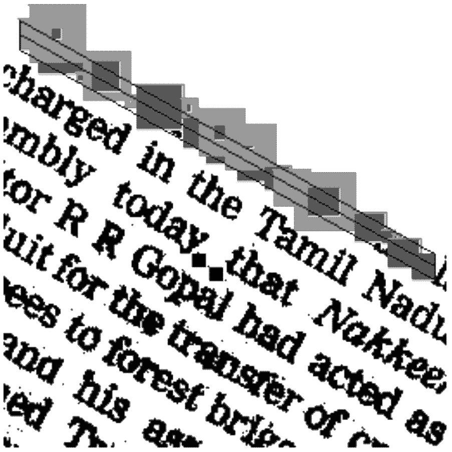


Figure 7. Document scanned at 200 dpi.

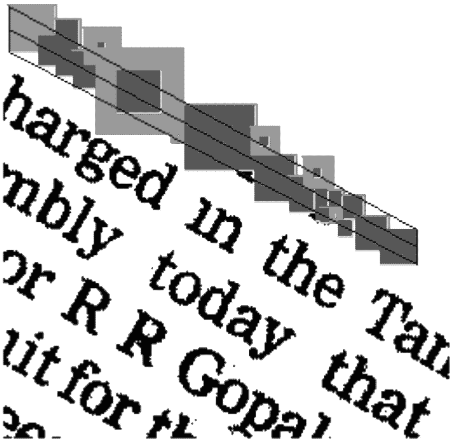


Figure 8. Document scanned at 300 dpi.

Table 1. Mean values of lowermost, uppermost and averaging methods compared among themselves.

True angle	Lowermost (mean)	Uppermost (mean)	Average (mean)
3	3·486	3·4770	3·4815
5	5·415	5·377	5·396
10	10·465	10·420	10·442
20	20·403	20·524	20·463
30	30·253	30·494	30·373

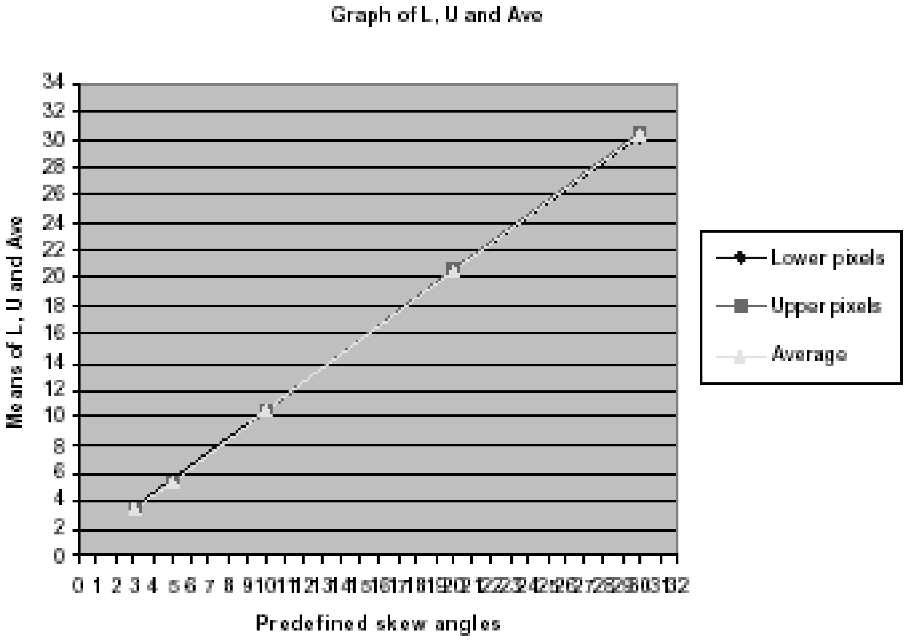


Figure 9. Graph of lowermost and uppermost methods with respect to table 1.

gives better results than other methods for some angles. However the method is sensitive to end characters of text lines. If the end characters are dots then the method fails to obtain the expected accuracy. This is shown in figures 11 and 12. With this, we conclude that the averaging method is better than the others.

The results of average method is compared with the results of other methods as shown in the table 3. Figure 11 and table 3 how that the proposed methods are very competitive and table 4 shows standard deviations and their graphs (figure 12), that indicate that the proposed method (dotted line and blue line) outperforms other methods in terms of obtaining consistent results. However, the method does not achieve the expected accuracy compared to Pal and Chaudhuri's (1996) method. These methods are robust to noise but computationally expensive compared to the proposed method.

Table 2. Comparative study of lowermost, uppermost and averaging methods with respect to standard deviations.

True angle	Lowermost (standard deviation)	Uppermost (standard deviation)	Average (standard deviation)
3	0·305	0·299	0·302
5	0·281	0·304	0·292
10	0·261	0·513	0·287
20	0·348	0·492	0·42
30	0·536	0·478	0·507

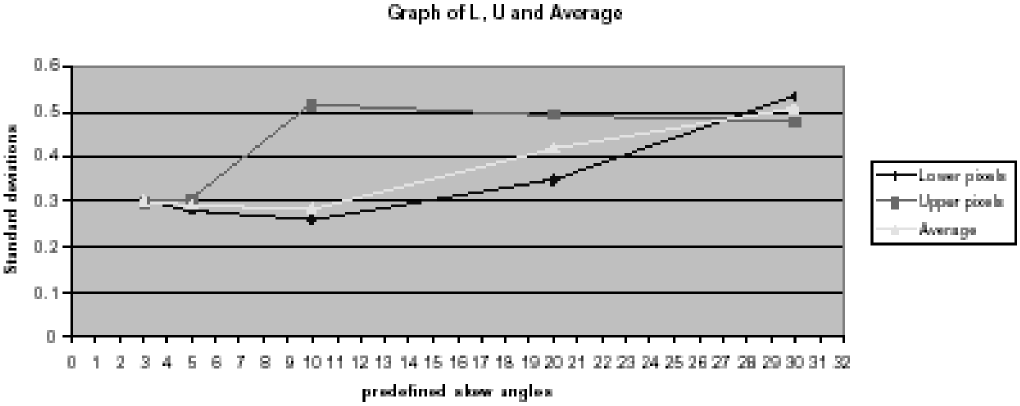


Figure 10. Graph of standard deviations with respect to table 2.

From tables 5 and 6 it is observed that the proposed techniques give consistent results for scaled and skewed documents also. Hence we conclude that the proposed techniques are invariant to scaling.

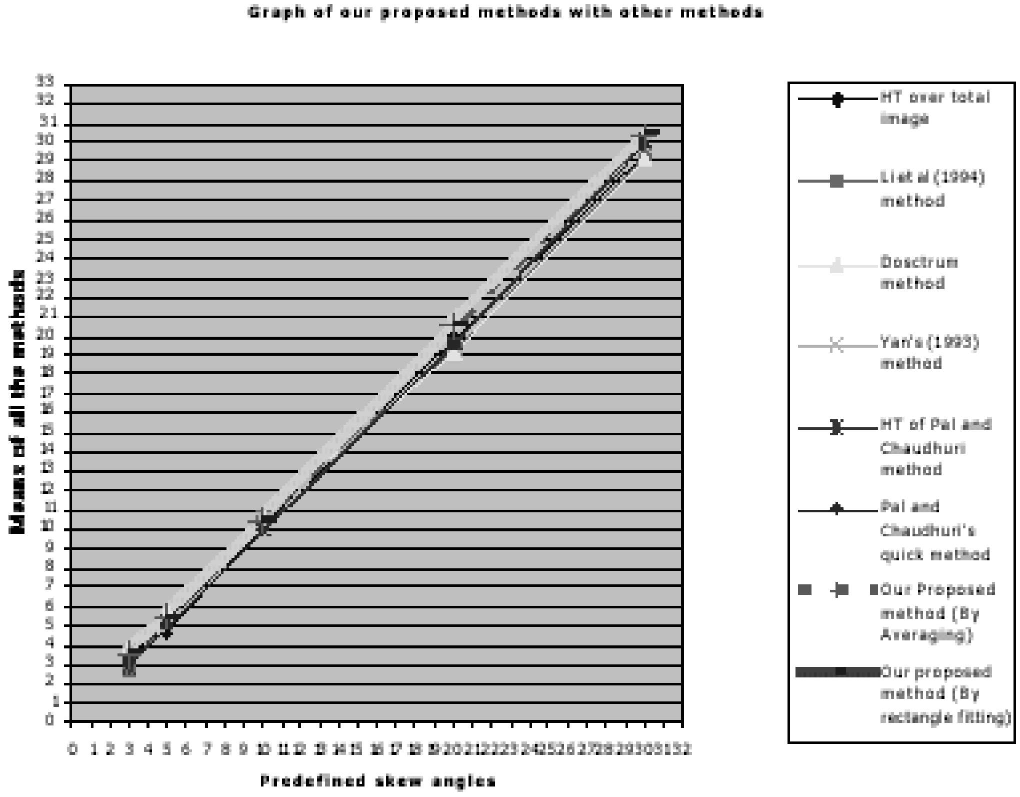


Figure 11. Graph of comparative study of proposed method with other known methods.

Table 3. Mean values of skew angles of 20 different images of different methods compared with mean of our proposed method.

True angle	A	B	C	D	E	F	G	H
3	3·72	2·80	3·82	3·69	2·88	3·14	3·4815	3·74
5	4·71	5·24	5·74	5·01	5·18	5·03	5·396	5·59
10	10·21	10·67	10·71	10·46	10·05	9·96	10·442	10·58
20	19·91	19·25	19·16	19·53	19·66	19·52	20·463	20·59
30	29·48	29·40	29·11	29·88	30·16	29·85	30·373	30·56

Table 4. The standard deviations of 20 different skew angles of different methods compared with the standard deviation of our proposed method.

True angle	A	B	C	D	E	F	G	H
3	0·079	0·534	0·627	0·384	0·228	0·331	0·302	0·608
5	0·155	0·687	0·295	0·489	0·436	0·269	0·292	0·336
10	0·733	0·840	0·997	0·717	0·216	0·233	0·287	0·366
20	0·777	1·013	0·422	0·918	0·366	0·349	0·42	0·365
30	0·559	0·965	0·665	0·537	0·313	0·379	0·507	0·462

A: Hough transform over total image; B: Hough transform on pixels selected by Le *et al* (1994) method; C: Docstrum method (O'Gorman 1993); D: Yan's (1993) method; E: Hough transform on pixels selected by Pal & Chaudhuri's (1996) method; F: Quick method of Pal and Chaudhuri (1996); G: Our proposed method (by averaging); H: Our proposed method (by rectangle-fitting)

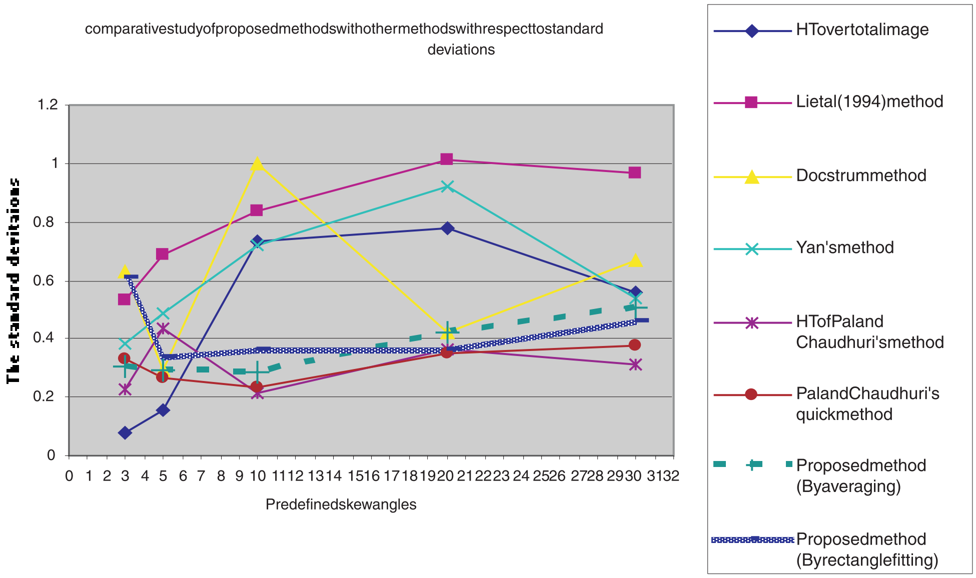


Figure 12. Comparative study of proposed method with other methods with respect to standard deviations.

Table 5. Skew angles for different scaled documents using averaging method.

True angle	SEA	75 dpi	100 dpi	150 dpi	200 dpi	300 dpi
3	3·4815	3·69	3·72	3·591	3·17	3·44
5	5·396	5·629	5·429	5·36	5·51	4·69
10	10·442	10·80	10·725	10·12	10·11	10·49
20	20·463	20·99	20·53	20·36	20·70	19·82
30	30·373	30·47	30·45	30·24	29·91	30·02

5. Results and discussions

In this paper, we have presented two new approaches for skew detection based on linear regression analysis and skew estimation based on rectangle-fitting for skewed text lines. These two methods are computationally inexpensive compared to the above HT based methods. Generally, if any method involves HT to estimate skew angles, it is considered computationally expensive according to the literature (Pal & Chaudhary 1996; Amin & Fischer 2003). This is because of the complexity of the HT. The complexity of the HT is in worst case $O(n\theta)$, where n is the number of coordinates and θ is the angular resolution, which runs from -90 to $+90$. If we consider 1° angular resolution then the complexity of the HT becomes $O(180n)$. If we consider 0·5° angular resolution then the complexity becomes $O(360n)$. If we assume θ is n in worst case then it becomes $O(n^2)$, however, the angular resolution is very important factor in deciding the number of computations and the complexity of the method. Similarly, the complexity of our proposed method based on linear regression analysis is $O(n)$, where n is the number of coordinates. Also the complexity of our proposed method based on rectangle-fitting for text line is $O(k)$, where k is a constant which represents the number of text lines present in the skewed document.

From the above discussion, we can conclude that our proposed methods are computationally inexpensive compared to HT-based methods. Similar inference can be drawn from Pal & Chaudhary (1996). In this paper, we observe that 515·16 s are required to estimate skew angle when we employ HT for whole image of size 512 × 512, 35·41 s when we apply HT-based method by Le *et al* (1994). A time of 30·07 s is required when we employ the HT-based method by Pal & Chaudharys (1996) and 10·35 s when we employ their method that does not involve HT. From these data, it is easily seen that methods involving HT are

Table 6. Skew angles for different scaled documents with rectangle-fitting method.

True angle	SERT	75 dpi	100 dpi	150 dpi	200 dpi	300 dpi
3	3·74	4·133	3·650	2·87	3·86	4·00
5	5·59	5·48	5·64	5·04	5·79	5·88
10	10·58	10·86	10·70	10·10	10·83	10·93
20	20·59	20·416	20·526	20·61	20·88	20·86
30	30·56	29·843	30·205	30·50	30·76	30·69

computationally expensive compared to other methods that do not involve HT. Hence, we conclude that our proposed methods are computationally inexpensive compared to HT-based methods. However, our proposed methods are not robust to noise and are not as accurate as the HT-based methods and Pal & Chaudhary (1996) quick method. This can be seen in figure 11. The proposed methods fail to achieve the expected accuracy if there is noise in the document, since the methods work based on the assumption that the space between the text lines is greater than the space between the words and characters. Noise may reduce the space between text lines, words or characters. Hence, the proposed methods are not robust to noise. This drawback can be overcome by modifying the proposed method by defining height and width of connected components to eliminate noise. The docstrum method (O' Gorman, 1993) is less accurate for almost all skew angles. Yan's (1993) method is quite accurate but is computationally time-consuming. Also, for its implementation some a priori knowledge about the spacing is needed to compute the correlation. The study made in this paper reveals that our proposed methods could be easily extended to achieve expected accuracy with minimum expense for complex documents containing half tone images, graphics and drawings.

The authors acknowledge the support extended by Vidya Shankar. One of the authors (S P) wishes to thank the AICTE, for a fellowship.

References

Amin A, Fischer S 2003 A document skew detection method using the Hough Transform. *Pattern analysis and applications* (London: Springer-Verlag) pp 243–253

Baird H S 1987 The skew angle of printed documents. *Proc. Soc. Photogr. Sci. Eng.* 40: 21–24

Caprari R S 2000 Algorithm for text page up/down orientation determination. *Pattern Recogn. Lett.* 21: 311–317

Duda R O, Hart P E 1973 *Pattern classification and scene analysis* (New York: Wiley-Interscience)

Gatos B, Papamarkos N, Chamzas C 1997 Skew detection and text line position determination in digitized documents. *Pattern Recogn.* 30: 1505–1519

Gonzalez R C, Woods R E 2000 *Digital image processing* (Reading, MA: Addison-Wesley)

Hashizume Yeh P S, Rasenfeld A 1986 A method of detecting the orientation of aligned components. *Pattern Recogn. Lett.* 4: 125–132

Hinds S C, Fisher J L, Amato D P 1990 A document skew detection method using run-length encoding and the Hough Transform. *In Proc. Int. Conference on Pattern Recogn.* 1: 464–468

Le D S, Thoma G R, Weehsler H 1994 Automatic orientation and skew angle detection for binary document images. *Pattern Recogn.* 27: 1325–1344

Liolios N, Fakotakis N, Kokkinakis G 2002 On the generalization of the form identification and skew detection problem. *Pattern Recogn.* 35: 253–264

O' Gorman L 1993 The document spectrum for page layout analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 15: 1, 162–171, 173

Pal U, Choudhuri B B 1996 An improved document skew angle estimation technique. *Pattern Recogn. Lett.* 17: 899–904

Postl W 1986 Detection of liner oblique structure and skew scan in digitized documents. *In Proc. Int. Conf. on Pattern Recognition*, pp 687–689

Shivakumara, Guru D S, Hemantha Kumar G, Nagabhushan P 2001 Document image mosaicing: A novel technique based on pattern matching approach. *Proceedings of the National Conference on Recent Trends in Advanced Computing (NCRTAC-2001)*, Tamil Nadu, pp 01–08

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2002a Skew detection in binary document image using linear regression analysis. National Conference. Proceedings of National Conference on Advanced Computer Applications – NCACA-2002, Dept. of Computer Science, NGM College, Pollachi, Coimbtore, Tamil Nadu, October

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2002b Text-skew detection through contour following in document image. *Proc. National Workshop on Computer Vision, Graphics and Image Processing – WVGIP-2002*, pp 39–44

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2003a Skew detection in binary text document images based on statistical analysis useful for document image mosaicing. *Proc 5th Annu. Conf. on Statistics, Computer and Applications*, New Delhi

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2003b Skew estimation of binary document images using static and dynamic thresholds useful for document image mosaicing. *Proc. National Workshop on IT Services and Applications, WITSA – 2003*, Jamia Milia Islamia, New Delhi

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2003c Skew estimation of binary document images: an approach based on text line boundary fitting. *Proc. National Seminar on Algorithms and Artificial Systems*, University of Madras

Shivakumara P, Guru D S, Hemantha Kumar G, Nagabhushan P 2003d Statistical methodology for skew detection in binary text document images for document image mosaicing. *J. Soc. Stat. Comput. Appl.* 1: 81–90

Srihari S N, Govindaraju V 1989 Analysis of textual images using the Hough Transform. *Machine Vision Appl.* 2: 141–153

Yan H 1993 Skew correction of document images using interline cross-correlation. *Comput. Vision, Graphics Image Process.* 55: 538–543