Available online at www.sciencedirect.com
SCIENCE DIRECT®
Pattern Recognition Letters 25 (2004) 1215–1229
Pattern Recognition Letters
www.elsevier.com/locate/patrec

A new algorithm for skew detection and correction

Rajiv Kapoor [a,*], Deepak Bagai [b], T.S. Kamal [c]

[a] *Computer Engineering, Netaji Subhas Institute of Technology, Azad Hind Fauz marg. sec-3, Dwarka, New Delhi 110045, Chandigarh, India*
[b] *Punjab Engineering College, Chandigarh, India*
[c] *SLIET, Longowal, Punjab, India*

Received 5 June 2003; received in revised form 28 November 2003
Available online 6 May 2004

Abstract

Here we have proposed two algorithms. The first one detects the skewing of words and the second corrects the skewing from handwritten words. Both algorithms make use of the Radon transform based projection profile technique. The method does not require pre-processing and it works equally good even with noise. The method is fast. The algorithms have been tested on words taken from more than 200 writers and the results obtained confirm the overall accuracy of proposed system. No error was detected.

Keywords: Character recognition; Skew detection

1. Introduction

A major application of pattern recognition is the optical character recognition (OCR). Numerous systems have been proposed during the last decades. Offline handwritten character recognition is a process where the computer understands automatically the image of a handwritten script. Offline handwriting is therefore distinguished from online handwriting, where a device such as a digitizing tablet measures the path of the pen. Online text capturing process provides more features as compared to the offline process but most of the applications must have offline capturing. Moreover, the age-old documents lying in offices have to be fed into the computer in text form. A number of applications can be developed including document transcription, automatic mail routing, and machine processing of forms, cheques and faxes (Park and Lee, 1996; Kim and Kim, 1996; Senior and Robinson, 1998; Gonzalez and Woods, 1993). This work has been carried out to make the Deva-Nagari text skew-free. Moreover, the handwritten material is generally not in the straight line; hence, it is extremely important to straighten the written word. The skew-detection system can be divided into the sections shown in Fig. 1. The processing starts with data acquisition and ends up with the skew-correction process.

* Corresponding author. Tel.: +91-1722737221; fax: +91-1125099029.
E-mail address: rajivkapoor@nsit.ac.in (R. Kapoor).

0167-8655/$ - see front matter © 2004 Published by Elsevier B.V.
doi:10.1016/j.patrec.2004.03.020

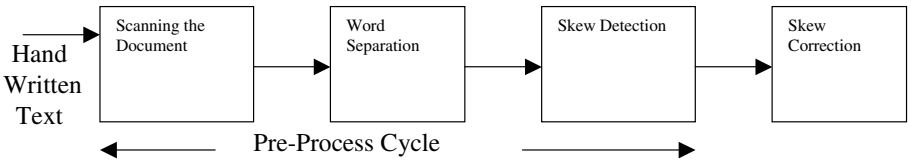


Fig. 1. Character recognition process flow diagram.

To capture the data from a handwritten document, a conventional flatbed scanner has been used. Subsequently a series of image-processing operations are carried out to normalize the image, so as to make it invariant to distortion (Kim and Kim, 1996). The writer used a black dot-pen for writing. Each sheet was scanned at 300 dpi resolutions to create a file. Once the skew has been detected and removed thereafter, a simple projection-based segmentation algorithm (Senior and Robinson, 1998) can be used to separate identifiable characters. In the next section, the concerned literature on the subject has been reviewed. However, skew detection, as a part of pre-processing stage is necessary, both for the segmentation and for the character recognition as well, since skewed words may cause considerable difficulties in both tasks. In this paper we present algorithm that deals with this important skew detection problem. The algorithm detects the skewing of printed or handwritten words and then removes the skew from a handwritten word. The method is based on the Radon transform. [1] The beauty of the method is that the skew of the whole document can be set right within few seconds. In the following sections, we present literature survey (Section 2) and Radon transform in Section 3. The proposed algorithms have been discussed in Sections 4, while the experimental results and the conclusion are given in Section 5. Once the skew is found, the document can also be separated into lines by taking Radon transform at same skew angle. The projection gives the distance of the Radon transform with line detected at various distances from the origin. In the similar manner taking the Radon transform at 90° to the detected skew angle can separate the words.

[1] http://eivind.imm.dtu.dk/staff/ptoft/Radon/Radon.html.

2. Existing methods

Kim and Govindaraju (1997) have employed chain code method of image representation. Chain code is a linear structure that results from quantization of the trajectory traced by the centers of adjacent boundary elements in an image array. Each data node in the structure represents one of eight grid nodes that surround the previous data node. Tracing of chain code components was carried out slant angle correction using a pair of one-dimensional filter. Each filter is a five-element array of different weights. Coordinates of the start and end points of each vertical line extracted provide the slant angle. Global slant angle is the average of all the angles of the lines, weighted by their length in the vertical direction since the longer line gives more accurate angle than the shorter one. Another technique depends upon the projection profile of the handwritten document where the horizontal or vertical profile is only a histogram of the information pixels scanned horizontally or vertically (Hou, 1983). For a script with horizontal text lines, the horizontal projection profile will have peaks at text line position and troughs at positions in between successive text lines. The projection profile taken in such a manner is computed at a number of angles. And for each angle, peak and trough height difference is measured. The maximum difference corresponds to the best alignment in the direction of the tilt, which in turn determines the tilt angle. Baird (1987) proposed a modification for quick convergence of this iterative approach. Akiyama and Hagita (1990) described an approach where the document is partitioned into vertical strips. The horizontal projection profiles are calculated for each strip and the angle of tilt is determined from the correlation of the profiles of the neighboring strips. The method is fast but less accurate. Pav-

lidis and Zhou (1992) proposed a method, which is based on vertical projection profile of horizontal strips, which works well when the angle of tilt is small. Yan (1993) proposed another method, which uses cross-correlation between the lines at a fixed distance. An analysis that shows projection profile methods are suitable for the tilt angles of less than 10°. Certain techniques based on Hough transform are also popular to detect the angle of tilt (Hinds et al., 1990; Le et al., 1994; Srihari and Govindraju, 1989). Hinds et al. (1990) modified the Hough transform method to reduce the amount of data to be processed. Le et al. (1994) found connected components in a script and considered only the bottom pixels of each component for Hough transform, which reduces the amount of information data. Pal and Choudhury (1996) have improved this approach. Fourier transform-based methods have also been used to detect the angle of tilt. Postl (1986) suggests an approach, where the direction for which the density of Fourier transform is the largest gives an estimate of the angle of tilt. Hashizume et al. (1986) propose the nearest neighbor clustering approach, to detect the angle of tilt. He found all the connected components in the document and computed the direction of its nearest neighbor for each component. Thereafter, a histogram of the direction angle is computed. Its peak indicates the document angle of tilt. O'Gorman (1993) suggested 'docstrum analysis'. These approaches are not limited to any range of tilt angle. It is clear that the component nearest clustering methods will not work well for DevaNagari script because characters are connected by the headline. Modified Hough transform method is also unsuitable for DevaNagari script because of the connecting nature of a word. Some pixels at the bottom of each word will make the data very sparse and its peak will be quite flattened. Now, from above, it is clear that the angle of tilt can be found easily only if some inherent characteristic of the script can be used and this is always so. Based on this characteristic, Chaudhury and Pal (1997) suggested a skew detection approach for the printed characters of DevaNagari script. This approach is based upon detecting the inherent feature (head line) in the DevaNagari word. Here, the author has detected the word in a skewed document by using the method of connected component labeling. Then, the upper envelope of the selected components is isolated. Finally, a new approach has been used to detect the skew based on the detection of digital straight line (DSL) segments from the upper envelope. DSL (digital arc) have been represented by chain code. This method was applied to printed characters. The documents were tilted by an angle ranging between 0° and ±45°. Minimum average execution time is 17.8 s. The angular resolution used in Hough transform is 1°. Another technique has been proposed (Kapoor et al., 2002), where again the Shiro-Rekha has been exploited but here the top cover of DevaNagari Word is extracted and the differential of that gives the angle of skew. This method is faster comparatively (Chaudhury and Pal, 1997) but lacks in terms of extra preprocessing. In this paper by Lehal and Dhir (1999) a range free skew detection technique for machine printed Gurmukhi documents has been presented. This approach can easily be extended to other Indian language scripts such as Devanagari and Bangla. Most characters in these scripts have horizontal lines at the top called headlines. The characters forming a word are joined at top by headlines, so that the word appears as one single component with headline. The ratio of pixel density above and below the headline of any word in Gurmukhi script is always less than 1. These inherent characteristics of the script have been employed and a new algorithm based on projection profile method has been devised. By inspecting horizontal and vertical projections at different angles in range $[0°, 90°]$, the skew angle of the document in range $[-180°, 180°]$ can be determined. Thus this approach is not limited to any range of skew angle and skewness in any document with orientation portrait or landscape and placed at any angle can easily be detected and removed. In this paper (Pal et al., 2001) they discussed about the documents having multi-skew text of DevaNagari and Bangla scripts. The connected components have been labeled and then selected. The upper layer of the selected components is found by the column-wise scanning from the top of the component. Portions of the upper envelop which look similar to DSL are detected. They are then

clustered into groups belonging to single text lines. Estimates from these individual clusters give the skew angle of each text line. Uchida et al. (2001) suggested a non-uniform slant correction technique where the slant correction problem has been formulated as an optimal estimation problem of local slant angles at all horizontal positions. The optimal estimation is bound by several constraints for the global and local validity of the local angles. The best-suited angle as per the constraints is searched. Our proposed method also exploits the inherent Shiro-Rekha (property) of the Deva-Nagari script and it does skew detection and correction of a document in a single attempt without any intermediate step. It is comparatively much efficient.

3. Radon/Hough transform

3.1. Introduction

The Radon transform can be defined as

$$\hat{g}(\rho,\tau)=\int_{-\infty}^{\infty} g(x,\rho x+\tau)\,\mathrm{d}x \tag{3.1}$$

where $\hat{g}(\rho,\tau)$ is a Radon transform of a continuous two-dimensional function $g(x,y)$. The Radon transform can be found by integrating values of $g(x,y)$ along slanted lines. The location of line is determined from the line parameters slope (ρ) and offset (τ). In principle the two parameters do not have limits but still discrete implementation will use a limited number of samples in both parameters in order to improve the computation.

Using the delta function, the Radon transform can be written as

$$\hat{g}(\rho,\tau)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\delta(y-\rho x-\tau)\,\mathrm{d}x\,\mathrm{d}y \tag{3.2}$$

The values of $\hat{g}(\rho,\tau)$ are function in two-dimensional (p,τ)-space. It is known that only a subset of functions can be Radon transformed analytically therefore a discrete approximation to the Radon transform that transforms a digital image is important. The easy way to sample the four variables linearly so as to work with only limited number of sampling parameters.

$$\begin{aligned} x &= x_m = x_{\min}+m\Delta x, \quad m=0,1,\ldots,M-1\\ y &= y_n = y_{\min}+n\Delta y, \quad n=0,1,\ldots,N-1\\ \rho &= \rho_k = \rho_{\min}+k\Delta\rho, \quad k=0,1,\ldots,K-1\\ \tau &= \tau_h = \tau_{\min}+h\Delta\tau, \quad h=0,1,\ldots,H-1 \end{aligned} \tag{3.3}$$

Here the $x_{\min}$ is the position of the first sample. Δx is the sampling distance of x and m is the discrete index used to number the M samples of x. Let us suppose that sampling of image $g(x,y)$ gives a digital image $g(m,n)$ and similarly the Radon transform of discrete image can be expressed as

$$\breve{g}(k,h)=\breve{g}(p_k,\tau_h). \tag{3.4}$$

Eq. (3.4) requires samples not found in the digital image, because linear sampling of all variables implies that $p_k x_m+\tau_h$ in general never coincides with the samples y_n. This problem can be corrected by using a nearest neighborhood approximation in the y-direction. Let us write n in a very simple manner. Where

$$\begin{aligned} n &= \left[\frac{P_k x_m+h\Delta\tau+\tau_{\min}-y_{\min}}{\Delta y}\right]\\ &= \alpha m+\beta \quad \text{where} \begin{cases} \alpha=\dfrac{x_m\Delta x}{\Delta y}\\ \beta=\dfrac{\rho_k x_m+\tau_h-y_{\min}}{\Delta y} \end{cases} \end{aligned} \tag{3.5}$$

$$\begin{aligned} \hat{g}(\rho_k,\tau_h) &= \int_{-\infty}^{\infty} g(x,\rho_k x+\tau_h)\,\mathrm{d}x\\ &\approx \Delta x\sum g(x_m,\rho_k x_m+\tau_h) \end{aligned} \tag{3.6}$$

Sampling of the function gives the digital image, which can be expressed as below:

$$g(m,n)=\hat{g}(x_m,y_n) \tag{3.7}$$

So to clear from the indices whether the continuous or the discrete version of the Radon transform is to be used, we will follow the notation as given below:

$$\hat{g}(k,h)=\hat{g}(\rho_k,\tau_h) \tag{3.8}$$

Hence the discrete Radon transform can be represented as a digital image. Linear Radon trans-

form can be defined in a more general form. There are three degrees of freedom, then the normal Radon transform is

$$\hat{g}(\xi_0,\xi_1,\xi_2)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y) \times \delta(\xi_0-\xi_1 x-\xi_2 y)\,\mathrm{d}x\,\mathrm{d}y \tag{3.9}$$

Eq. (3.9) describes the line with three degree of freedom. The three parameters should always have a link, which eliminates one degree of freedom. Here the parameters are given as in Eq. (3.10).

$$(\xi_0,\xi_1,\xi_2)=(-\tau,\rho,-1) \tag{3.10}$$

Another form of Radon transform is used in many fields like medical imaging, where the fundamental function $g(x,y)$ has no preferred orientation and this leads to describing the line in its normal form as:

$$\rho=x\cos\theta+y\sin\theta \tag{3.11}$$

where $(\xi_0,\xi_1,\xi_2)=(\rho,\cos\theta,\sin\theta)$ and $0\leqslant\theta<\pi$ and $-\rho_{\max}\leqslant\rho\leqslant\rho_{\max}$.

We also understand that a point source gets transformed into a sinusoid in the parameter domain.

$$\begin{aligned} &g(x,y)=0 \quad \text{for } \sqrt{x^2+y^2}>\rho_{\max} \\ &\quad\Rightarrow \hat{g}(\rho,\tau)=0 \quad \text{for } |\rho|>\rho_{\max} \end{aligned} \tag{3.12}$$

Again a discrete approximation will require all parameters sampled linearly. Eq. (3.12) implies that the function $g(x,y)$ should be shifted towards the center of the coordinate system, in order to get the smallest number of samples in the ρ-direction.

$$\begin{aligned} x&=x_m=x_{\min}+m\Delta x, \quad m=0,1,\ldots,M-1 \\ y&=y_n=y_{\min}+n\Delta y, \quad n=0,1,\ldots,N-1 \\ \theta&=\theta_t=\theta_{\min}+t\Delta\theta, \quad t=0,1,\ldots,T-1 \\ \rho&=\rho_r=\rho_{\min}+r\Delta\rho \quad r=0,1,\ldots,R-1 \end{aligned} \tag{3.13}$$

And if the image is square then

$$\Delta x=\Delta y$$
$$M=N$$

which implies that the samples should lie in a symmetrical interval around zero, i.e.,

$$x_{\min}=-x_{\max}=-\frac{(M-1)}{2}\Delta x$$
$$y_{\min}=x_{\min}=-y_{\max}=-\frac{(M-1)}{2}\Delta x$$
$$\rho_{\min}=-\rho_{\max}=-\frac{(R-1)}{2}\Delta\rho$$

Considering the angular sampling, the angular staring point can be chosen as

$$\theta_{\min}=0$$

And the sampling interval of θ should be set to span π, i.e.,

$$\Delta\theta=\frac{\pi}{T}$$

3.1.1. Nearest neighbor interpolation

Let there be a digitally sampled image $g(m,n)$. Eq. (3.4) requires samples not found in the digital image, because linear sampling of all variables implies that $p_k x_m+\tau_h$ in general never coincides with the samples y_n. This problem can be corrected by using a nearest neighborhood approximation in the y-direction. So the Radon transform can be approximated by the discrete Radon transform. The constant term Δx can be neglected. This term is desired for continuous Radon transform as given in Eq. (3.14).

$$g(k,h)=\Delta x\sum_{m=0}^{M-1} g(m,n(m;k,h))$$

where

$$n(m;k,h)=\left[\frac{p_k x_m+\tau_h-y_{\min}}{\Delta_y}\right] \tag{3.14}$$

Another problem arises because discrete point $(m,n(m;k,h))$ need not lie within the finite image. If the point lies outside the image the value needed could be set to zero. The time consuming part of the loop i.e. m-loop optimization should be done very carefully. So calculate values of $m_{\min}$ and $m_{\max}$ as given in Eq. (3.15).

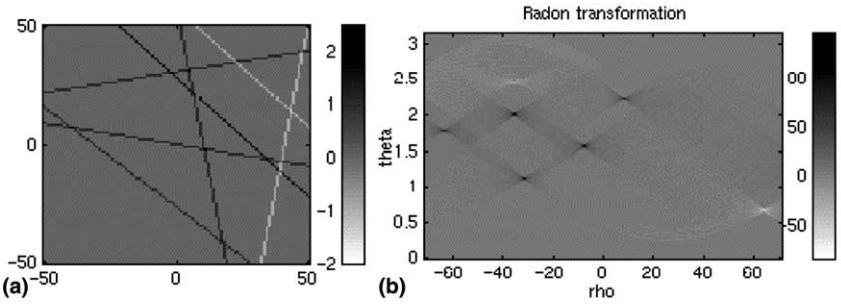


Fig. 2. (a) Input image with noisy background, (b) Radon transform of (a).

Table 1
Line parameters corresponding to Fig. 2

Line number	1	2	3	4	5	6	7
ρ	10	20	40	−20	0	30	−40
θ	0.175	0.785	0.785	0.873	1.396	1.745	2.967
μ	1	1.5	−1	1	1	1	−1

$$0 \leqslant n = [\alpha m + \beta] \leqslant N-1 \Rightarrow \begin{cases} \dfrac{-(\beta+0.5)}{\alpha} \leqslant m < \dfrac{N-0.5-\beta}{\alpha} & \text{if } \alpha > 0 \\ \dfrac{N-0.5-\beta}{\alpha} \leqslant m < -\dfrac{\beta+0.5}{\alpha} & \text{if } \alpha < 0 \end{cases}$$
$$\Rightarrow \begin{cases} m_{\min} = \max\left\{0, \left\lceil -\dfrac{\beta+0.5}{\alpha} \right\rceil\right\} \text{ and } m_{\max} = \min\left\{M-1, \left\lfloor \dfrac{N-0.5-\beta}{\alpha} \right\rfloor\right\} & \text{if } \alpha > 0 \\ m_{\min} = \max\left\{0, \left\lceil -\dfrac{N-0.5-\beta}{\alpha} \right\rceil\right\} \text{ and } m_{\max} = \min\left\{M-1, \left\lfloor \dfrac{\beta+0.5}{\alpha} \right\rfloor\right\} & \text{if } \alpha < 0 \end{cases} \tag{3.15}$$

Fig. 2 below shows an image with 101×101 pixels, where seven lines can be found with parameters as given in Table 1. The parameter domain representation of the Radon transform has been shown in Fig. 2(b) using nearest neighbor approximation. Seven peaks are detectable which correspond to the curve amplitudes μ. More number of samples are required in the parameter domain as it seems for original image (Table 2). Sparse sampling will not result in detection of all lines.

3.1.2. The Hough transform

The Hough transform maps the individual pixels from the image domain into the parameter domain.

$$\hat{g}(\rho,\tau) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x^*,y^*)\delta(\tau - y^* + \rho x^*)\,\mathrm{d}x^*\,\mathrm{d}y^* \tag{3.16}$$

Table 2
Sampling parameter settings corresponding to Fig. 2

Image domain		Parameter domain	
Parameter	Value	Parameter	Value
M	101	R	201
N	101	T	315
Δx	1	$\Delta\rho$	0.707
Δy	1	$\Delta\theta$	0.01
$x_{\min}$	−50	$\rho_{\min}$	−70.7
$y_{\min}$	−50	$\theta_{\min}$	0

For each point in the image $g(x^*, y^*)$, draws a line in the transformed plane. The line is found by setting the argument of the last delta function to zero. Hough transform was implemented using a nearest neighbor approximation in the parameter domain. It has been mentioned above that dense sampling is desirable in case of Radon transform and exactly opposite to this, sparse sampling results in better results in case of the Hough transform. Same image containing six lines with a background noise cannot be taken for the Hough transform. Hence Background is removed and sparse sampling of the image gives equally good results. In case of skew detection, the digital image will have huge sample database and this will result in inferior results. In case of Hough τ (offset) has been expressed in terms of x^* and y^* so that this may result in a line in parametric domain for a single point in the image domain. The other parameters have also been given in Eq. (3.17).

$$\left.\begin{array}{ll} \tau = y^* - \rho x^* & \\ x = x_m = x_{\min} + m\Delta x & m = 0, 1, \ldots, M-1 \\ y = y_n = y_{\min} + n\Delta y & n = 0, 1, \ldots, N-1 \\ \rho = \rho_k = \rho_{\min} + k\Delta\rho & k = 0, 1, \ldots, K-1 \\ \tau = \tau_h = \tau_{\min} + h\Delta\tau & h = 0, 1, \ldots, H-1 \end{array}\right] \tag{3.17}$$

Here the $x_{\min}$ is the position of the first sample, Δx the sampling distance of x, m is the discrete index used to number the M samples of x.

$$\begin{aligned} \tau = y - \rho x \Rightarrow h & \\ &= \left[\frac{y_n - (k\Delta\rho + \rho_{\min})x_m - \tau_{\min}}{\Delta\tau}\right] \\ &= \kappa k + \zeta \quad \text{where} \begin{cases} \kappa = -\dfrac{x_m \Delta\rho}{\Delta\tau} \\ \zeta = \dfrac{yn - x_m\rho_{\min} - \tau_{\min}}{\Delta\tau} \end{cases} \end{aligned} \tag{3.18}$$

Parameter h has been expressed in Eq. (3.18). If we consider the data as a square image then $\Delta\tau = \Delta y$ and $\tau_{\min} = y_{\min}$. The important result that Hough transform malfunctions when the parameter domain is sampled densely and works well with a coarsely sampled parameter domain, is understandable because the way the Hough transform is defined implies that all pixels lying around the digital line within a band of $\Delta\tau/\Delta y$, measured in pixels in the n-direction, will be mapped into the same point, (k, h), in the discrete parameter domain. And using a densely sampled parameter domain with $\Delta\tau = 0.1 * \Delta y$ implies that the width of the band in the image domain only amounts to 1/10 of a sample, hence a statistical point of view, only every tenth pixel along the digital line contributes to the discrete parameter domain and a coarsely sampled parameter domain with $\Delta\tau = 5 * \Delta y$ implies that for a given point in the discrete parameter domain, and a certain value of x_m, five pixels will contribute to the same position in the parameter domain, if their value is different from zero. If no prior knowledge of the image is there then the reasonable value of sampling distance of τ is $\Delta\tau = \Delta y$.

3.2. Implementation parameters selection

The Radon transform of a continuous function is found by integrating the values of $g(x, y)$ along started lines, two location of line is determined from to line parameters slope and object ρ and τ respectively. In principle the two parameters do not have limits though as it will be shown, discrete implementation will use a limited number of samples in both parameters directions. In Radon transform ρ and τ are used as the arguments. To reduce the computational cost the values of n in the Radon are limited to the $m_{\min}$ and $m_{\max}$, i.e. if the sampling of the parameter domain in the Radon transform is dense then the peak is broad and if the parameter domain sampling is sparse then important information might be lost. In the case of Hough transform, it is just the reverse of as in Radon transform. In sparse sampling of the parameter domain the more number of peaks are detected when compared to Radon transform. But the resolution in Hough transform is poor. Hough transform with the following parameters was tested and all programs were written in Matlab 6.1 running on P-IV, 1.7 GHz Dell Computer (Tables 3 and 4).

The Hough transform and Radon transforms are similar in the continuous case. For discrete case they are not identical. And the Hough transform is generally used in its discrete form.

Table 3
Sampling parameter settings

Image domain		Parameter domain	
Parameter	Value	Parameter	Value
M	101	H	101
N	101	K	101
Δx	1	$\Delta\rho$	1
Δy	1	$\Delta\tau$	1
$x_{\min}$	−50	$\rho_{\min}$	1
$y_{\min}$	0	$\tau_{\min}$	0

Fig. 3 shows the Hough transform taken with parameters in Table 5. This shows that Hough transform works well with the sparse sampling in the parametric domain and malfunctions if sampled densely in the parametric domain.

Fig. 3 shows that the Hough transform works well with sparse sampling. And if the sampling rate is kept constant still the Hough transform takes much more time for with increase in the size of the document image. It has been noticed from Table 6 and the Chart shown in Fig. 4 that even when sparse sampling is done still, the time taken by the Hough transform is much more than the time taken by Radon transform.

3.3. Complexity and the performance analysis

Radon transform works in line parameter extraction even in presence of noise. Fig. 5 explains when the parametric domain values have been taken from Table 3 for Hough transform so as to have best possible Hough transform and the Table 5 has been used for getting the Radon transform of Fig. 5(b) (a noisy image). Radon transform are able to transform each of the lines into peaks positioned corresponding to the parameters of the lines. In this way Radon transform converts a difficult global detection problem to local peak detection problem in the parameter

Table 5
Parameter table with dense parameter domain sampling

Image domain		Parameter domain	
Parameter	Value	Parameter	Value
M	101	H	101
N	101	K	101
Δx	1	$\Delta\rho$	0.004
Δy	1	$\Delta\tau$	0.15
$x_{\min}$	0	$\rho_{\min}$	1
$y_{\min}$	0	$\tau_{\min}$	0

Table 4
Line parameters corresponding to Fig. 3(a)

Line number	1	2	3	4	5	6
ρ	−0.33	−0.10	0.25	−0.25	0.50	0.00
τ	50	70	60	40	50	20
x_{start}	−41	−11	−50	−11	−50	−41
x_{end}	29	50	9	29	9	9

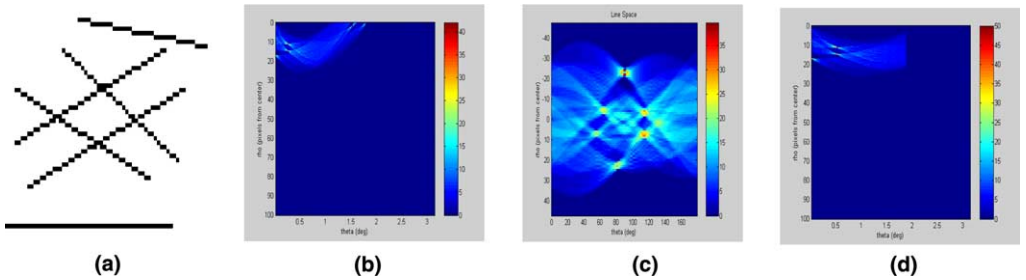


Fig. 3. (a) An image with six lines, (b) Hough transform, (c) Radon transform and (d) Hough transform with parameters from Table 5.

Table 6
Time taken by Hough and Radon transforms with parametric domain values taken from Table 5

Data	Hough	Radon	Difference	Data type
1	0.594	0.344	0.25	2 lines
2	0.734	0.203	0.531	3 lines
3	0.828	0.218	0.61	4 lines
4	3.578	0.297	3.281	Word Uski
5	7.39	0.344	7.046	Doc. English
6	1.594	0.234	1.36	6 lines
7	4.14	1.438	2.702	Blurred Doc
8	5.703	0.594	5.109	Hindi Doc

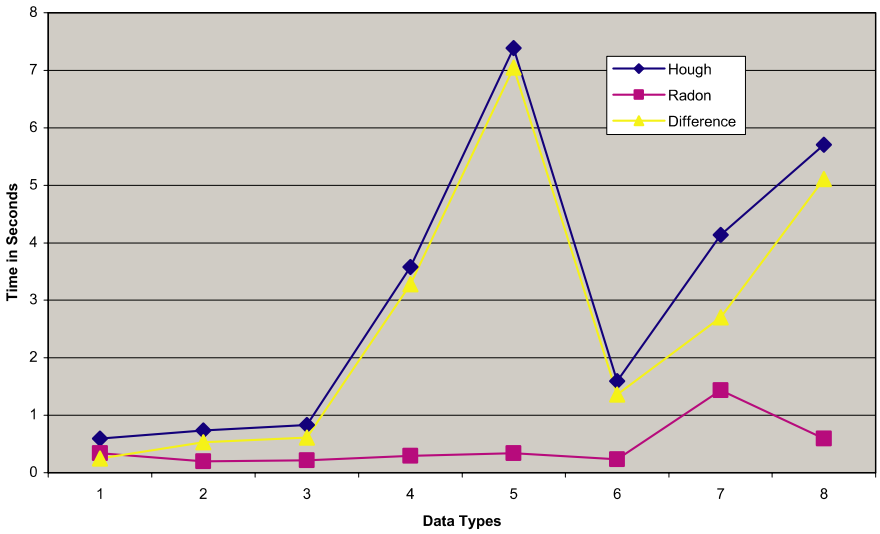


Fig. 4. Chart showing the difference in the time taken by the Hough and Radon transforms with parameters taken from Table 3.

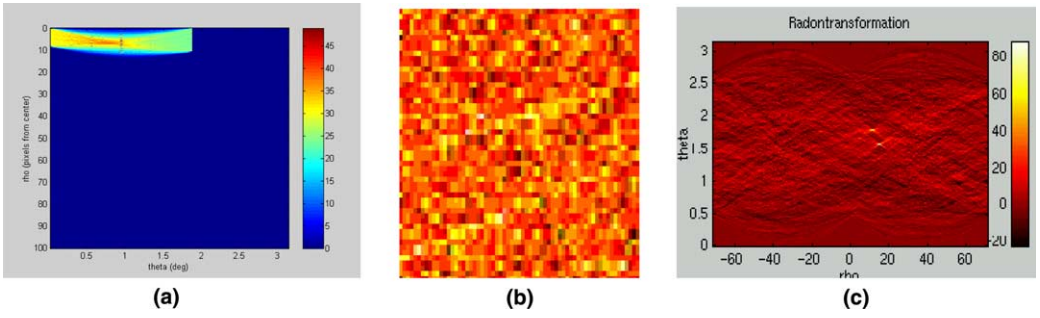


Fig. 5. (a) Hough transform of (b), (b) noisy image and (c) Radon transform of (b).

domain, and the actual parameters can be recovered by for e.g. threshold to Radon transform especially in this noisy case the Hough transform algorithm fail. An alternative is edge detection fitters, succeeded by a procedure for linking the individual pixels together; and finally to use linear regression for estimation of parameters. Algorithms of this kind have problems with intersecting

lines and in case of high noise level it is difficult to stabilize the edge detection filters.

3.3.1. Steep lines

When the images include one or more lines with high slope the parameters of the line having less slope will be identified and the parameters of the line having large slope, i.e. nearly vertical are not easily detectable. Fig. 6 shows that detected.

A way to overcome the described problem is to compute two parameter domains. Note that a discrete implementation of the Radon transform need not imply any additional programming by applying discrete Radon transform of the digital image. By using wavelet we can find the positive and negative valued images also.

3.3.2. Normal Radon transform

On this form the line is described with three degrees of freedom, which is one to many to describe a line. So the three parameters should always have a link, which removes one degree of freedom along with other four discrete parameters as given in Eq. (3.19).

$$\begin{aligned}
&(\xi_0, \xi_1, \xi_2) = (-\tau, \rho, -1) \\
&\left.\begin{array}{ll}
x = x_m = x_{\min} + m\Delta x & m = 0, 1, \ldots, M-1 \\
y = y_n = y_{\min} + n\Delta y & n = 0, 1, \ldots, N-1 \\
\theta = \theta_t = \theta_{\min} + t\Delta\theta & t = 0, 1, \ldots, T-1 \\
\rho = \rho_r = \rho_{\min} + r\Delta\rho & r = 0, 1, \ldots, R-1
\end{array}\right]
\end{aligned} \tag{3.19}$$

And if the image is square then

$$\Delta x = \Delta y$$
$$M = N$$

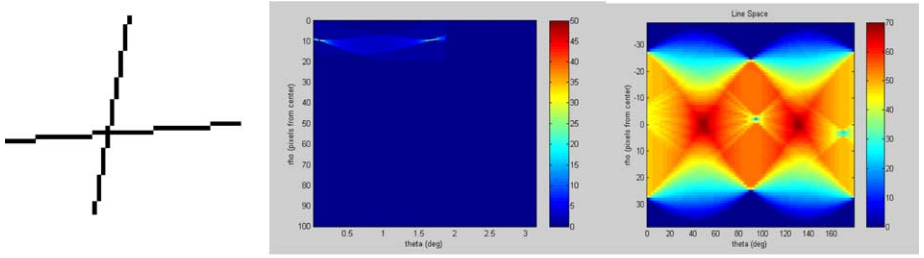
Other parameters have been given below for reference.

$$x_{\min} = -x_{\max} = -\frac{(M-1)}{2}\Delta x$$
$$y_{\min} = x_{\min} = -y_{\max} = -\frac{(M-1)}{2}\Delta x$$
$$\rho_{\min} = -\rho_{\max} = -\frac{(R-1)}{2}\Delta\rho$$
$$\theta_{\min} = 0$$
$$\Delta\theta = \frac{\pi}{T} \quad \text{for normalized Radon transform}$$

In case of line detection or any similar application while using Radon transform, the parameter domain must be sampled sufficiently dense, in order to assure that the discrete approximations to the Radon transform does not cause aliasing problems, therefore in normalized Radon transform also, nearest neighbor interpolation has been used. The number of computation required for Hough and the Radon transform have been given as below in Eq. (3.20).

$$\begin{aligned}
&O\ \text{Radon} = O(K \cdot H \cdot M) = O(M3) \\
&O\ \text{Hough} = O((MN)rk) = O((MN)rM)
\end{aligned} \tag{3.20}$$

The computational complexity of the sinc interpolation is 100 times higher than the other two methods as given in Table 7. The meaning of the normal parameters used to specify the position of the line are show in figure, the parameter is the shortest distance from the origin of the coordinate system to the line and θ is the angle corresponding to the angular orientation of the line. The parameters taken for comparing the three methods have been given in Table 7(b).



(a) (b) (c)

Fig. 6. (a) An image with two lines having a slope of 1/9 and 9, (b) the corresponding discrete parameter domain Hough transform, (c) corresponding Radon transform.

Table 7
Comparison of three interpolation methods (panel A) and corresponding parameters (panel B)

Panel A

Method	Time (s)
Nearest neighbor	2.9
Liner interpolation	4.2
Sinc. interpolation	1450

Panel B

Image domain		Parameter domain	
Parameter	Value	Parameter	Value
M	101	R	201
N	101	T	315
Δx	1	$\Delta\rho$	0.707
Δy	1	$\Delta\theta$	0.01
$x_{\min}$	−50	$\rho_{\min}$	−70.7
$y_{\min}$	−50	$\Delta\tau_{\min}$	0

3.3.3. Linear interpolation

It requires the sum of twice as many samples compared to nearest neighbor interpolation.

$$m_{\min} = \max\left\{0, \left\lceil -\frac{\beta}{\alpha}\right\rceil\right\} \quad \text{and}$$
$$m_{\max} = \min\left\{M-1, \left\lfloor \frac{N-1-\beta}{\alpha}\right\rfloor\right\} \quad \text{if } \alpha > 0 \tag{3.21}$$

Sampling properties of the discrete Radon transform should ensure that a given digital image the parameter domain must be sampled sufficiently dense in order to avoid aliasing problems.

$$\begin{aligned} 0 \leqslant n = [\alpha m + \beta] &\leqslant N-2 \\ \Rightarrow \begin{cases} -\dfrac{\beta}{\alpha} \leqslant m < \dfrac{N-1-\beta}{\alpha} & \text{if } \alpha < 0 \\ \dfrac{N-1-\beta}{\alpha} \leqslant m < \dfrac{\beta}{\alpha} & \text{if } \alpha > 0 \end{cases} \\ \Rightarrow m_{\min} = \max\left\{0, \left\lceil -\frac{N-1-\beta}{\alpha}\right\rceil\right\} &\quad \text{and} \\ m_{\max} = \min\left\{M-1, \left\lfloor -\frac{\beta}{\alpha}\right\rfloor\right\} &\quad \text{if } \alpha < 0 \end{aligned} \tag{3.22}$$

The parameters of a Radon transform of a discrete line has to be consider so that the peak is very broad, which suggests that the sampling of the parameter domain is unnecessary dense. If the parameters are sampled very sparsely then one of the images may vanish so the important information might be lost. Eqs. (3.21) and (3.22) suggest the values to be taken for $m_{\min}$ and $m_{\max}$.

Sampling of the parameter domain also depends on the tape of interpolation being used and type of image that using nearest neighbor approximation requires a dense sampling of the parameter domain in order to assure that the peak is found. By sparse sampling in the case of Radon transform malfunctions and only two peaks are clearly marked due to under sampling of the domain. Hough transform performs well. More peaks can be found but it should be noted that the resolution is poor. Radon transform is suited for line parameter extraction even in presence of noise. Table 8 gives the choice among various interpolation techniques.

3.3.4. Computational complexity

If the image only contains one non-zero pixel then Hough transform will be the faster of the two transforms. But more realistic value of $(M \cdot N)r$ lies between M and MN, depending on the structure of the image. More the data the more time-expensive the Hough transform becomes. The document data is enormous and a full-page document takes approximately an hour to detect the skew. This makes Hough transform not suitable for document skew detection. Using slope and offset parameters and nearest neighbor approximation, the two algorithms give exactly the same discrete parameter domain, when choosing $\Delta\tau = \Delta y$ and $\tau_{\min} = y_{\min}$. It is should be noted that the Hough transform needs a multiplicative factor of Δx in order to get the same scaling as the discrete Radon transform. A major difference of the two algorithms comes with the time needed to compute

Table 8
Time based comparison among various interpolation techniques

Method	Time (s)
Nearest neighbor	1.4
Nearest neighbor (optimized)	0.45
Liner interpolation	1.7
Sinc. interpolation	287

the discrete parameter domain. By implementing the linear interpolation to the Hough transform does not imply much change to the optimization strategy of computing the range of m values. Corresponding to the pixels of the image clearly marked peaks are found by use of the discrete Radon transform. By Hough transform the levels are very low and the regions of the expected lines are weakly marked.

3.4. Hough vs. Radon

Let us have a look at the implementation and performance study of both the transforms so that their application domain could be distinguishable. *Hough transform* maps the individual pixels from the image domain into a shape in the parameter domain where as in Radon transform a shape in the image domain into a single pixel in the parameter domain. A major difference of the two algorithms comes with the time needed to compute the discrete parameter domain. With a given digital document image the parameter domain must be sampled sufficiently dense in order to avoid aliasing problems. The parameters of a Radon transform of a discrete line has to be considered as if the peak is very broad, which suggests that the sampling of the parameter domain is unnecessary dense. If the parameters are sampled very sparsely then one of the images may vanish so the important information might be lost. Sampling of the parameter domain also depends on the type of interpolation being used and type of image used. Nearest neighbor approximation requires a dense sampling of the parameter domain in order to assure that the peak is found. By sparse sampling in the case of Radon transform peaks may not be clearly marked due to under sampling of the domain. Radon transform may malfunctions. Even if by adjusting the parameters, all peaks are found but it should be noted that the resolution would be very poor. Hence it is understandable that to get good results using Radon transform, the parameter domain has to be densely sampled and in case of Hough, it is just the reverse for getting equally good results. This has been found that the similar parameter domain sampling where a variety of images have been tried, Hough transform takes much more time as compared to the Radon transform. So from sampling parameters Radon transform gives better results compared to Hough transform. To further support the conclusion, let us study GRT and IPM as given below. The generalized Radon transform can be defined as

$$g(\xi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x,y)\hat{\delta}(\phi(x,y,\xi))\,\mathrm{d}x\,\mathrm{d}y \qquad (3.23)$$

3.4.1. Image point mapping

Estimation of discrete parameter domain

$$\hat{g}(j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g(m,n)\delta(n - \phi(m;j))$$

The above equation shows that image point (m,n) is transformed into a parameter curve where $n = \phi(m;j)$. Thus mapping of image points with zero value is needless. Note that IPM is a generalized Hough transform, which has been ex-

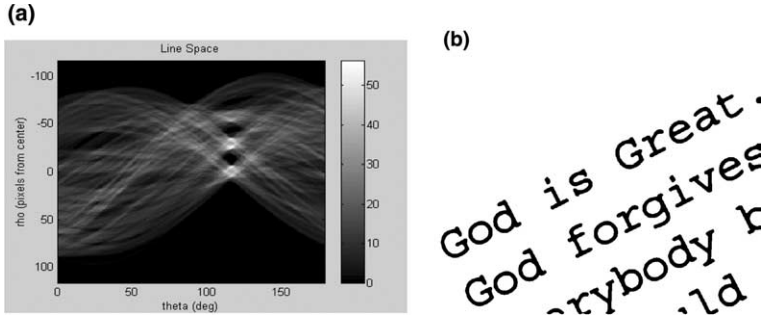


Fig. 7. (a) Radon transform of (b), (b) document having English script.

tended to handle a large set of non-linear transforms curves. One of the fundamental differences between the IPM and GRT is, that GRT requires rounding in the image domain, where as IPM rounds in the parameter domain. The computational complexities of GRT and IPM respectively are as given in Eq. (3.24).

$$O_{\mathrm{GRT}} = O\left(M\prod_{i=1}^{\eta} J_i\right) O_{\mathrm{IPM}} = O\left((M,N)_r \prod_{i=1}^{\eta-1} J_i\right) \tag{3.24}$$

where $(M,N)_r$ indicates that only a reduced number of image points are to be transformed. The cost to test the non-zero values is assumed to be negligible. GRT require a dense parameter domain sampling, while IPM gives rise to blurring in the parameter domain in the case of dense parameter sampling, but works well with a coarse sampling domain. Parameter Domain blurring achieved by use of GRT can be interpreted in a way that it makes it usable for parameter clustering. The direct use of GRT is computationally expensive. Lines with wiggles can be incorporated in the Radon transform and still it detects. If there exist additive noise in the image, is analyzed with respect to the curve parameter detection and here the Hough transform will not work in case of noise. It works equally well in case of document images containing non-DevaNagari script. Fig. 7 has been given as a sample.

4. Skew correction algorithm

Skew correction algorithm is shown in Fig. 8.

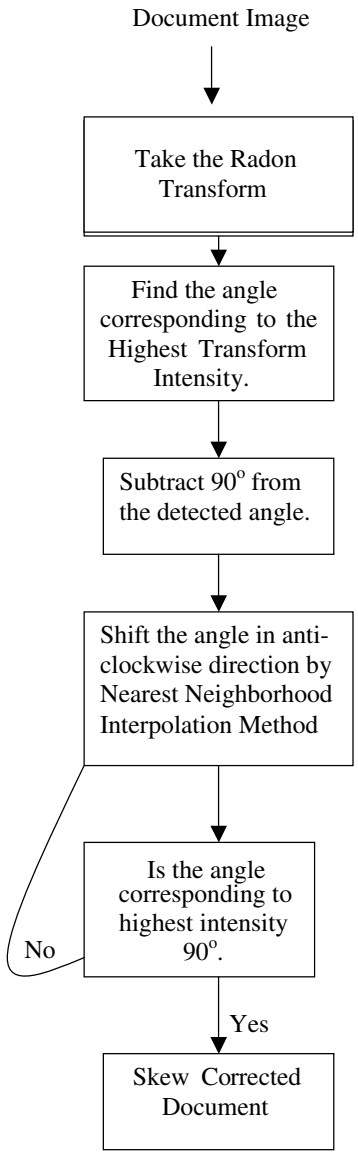


Fig. 8. Flow chart for skew correction.

5. Results and conclusion

Both algorithms have been tested with more than two hundred DevaNagari words taken from two hundred different writers. Moreover, the skew correction algorithm has also been tested on printed words. Figs. 9 and 10 show the process of skew removal.

The skew correction algorithm has dealt successfully with every word printed or handwritten with any skewing angle. Few results are presented above in Figs. 9 and 10. However, problem may appear in the case of words with non-uniform Shiro-Rekha or characters without Shiro-Rekha. In such cases, the above-proposed algorithm may not work. Though, the algorithm is insensitive to noise and hence the method works perfectly well

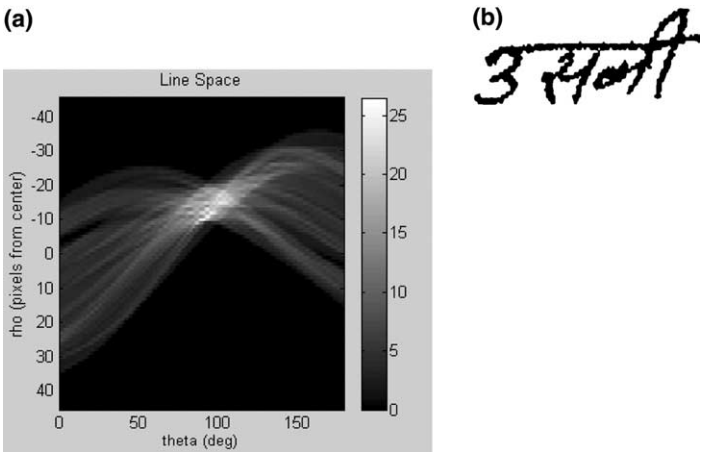


Fig. 9. (a) Radon transform of image at 0° skew, (b) corresponding image.

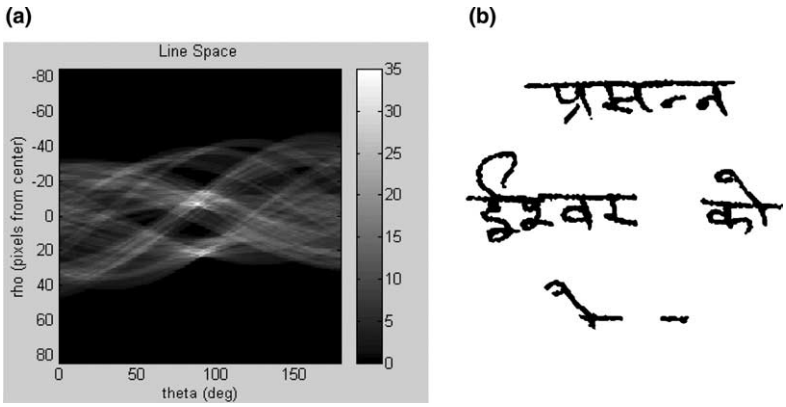


Fig. 10. (a) Radon transform of document at 0° skew, (b) corresponding image.

even with the characters written on paper having background template. The method is capable of detecting skew of the documents rotated at any angle. On the other hand, if the text is a only a word of very short length and the text also contains vertical line then the possibility for the short words-skew to be confused, is more since the skew angle is very small in a word of two characters and there is every possibility of detecting more than one line at different angles.

References

Akiyama, T., Hagita, N., 1990. Automatic entry system for printed documents. Pattern Recognition 23, 1141–1154.

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

Chaudhury, B.B., Pal, U., 1997. Skew angle detection of digitized Indian script documents. IEEE Trans. PAMI 19.

Gonzalez, R.C., Woods, R.E., 1993. Digital image processing reprinted with corrections, Person Education Asian September.

Hashizume, A., Yeh, P.S., Rosenfeld, A., 1986. A method of detecting the orientation of aligned components. Pattern Recognition Lett. 4, 125–132.

Hinds, S.C., Fisher, J.I., D'Amato, D.P., 1990. A document skew detection method using run-length encoding and the Hough transform. In: Proc. 10th Internat. Conf. on Pattern Recognition, 1. pp. 464–468.

Hou, H.S., 1983. Digital Document Processing. Wiley.

Kapoor, R., Bagai, D., Kamal, T.S., 2002. Skew angle detection of a cursive handwritten DevaNagri script character image. J. Indian Inst. Sci. Bangalore, India 82 (3&4), 161–176.

Kim, G., Govindaraju, V., 1997. Alexicon driven approach to handwritten word recognition for real-time applications. IEEE Trans. PAMI 19.

Kim, H.J., Kim, P.K., 1996. Recognition of off-line handwritten Korean characters. Pattern Recognition 29, 245–254.

Le, D.S., Thoma, G.R., Wechsler, H., 1994. Automatic page orientation and skew angle detection for binary document images. Pattern Recognition 27, 1325–1344.

Lehal, G.S., Dhir, R., 1999. A range free skew detection technique for digitized Gurmukhi script documents. ICDAR 20–22 (September), 147–152.

O'Gorman, L., 1993. The document spectrum for page layout analysis. IEEE Trans. PAMI 15, 1162–1173.

Pal, U., Choudhury, B.B., 1996. An improved document skew angle estimation technique. Pattern Recognition Lett. 17, 899–904.

Pal, U., Mitra, M., Chaudhuri, B.B., 2001. Multi-skew detection of Indian script documents. ICDAR, 292–296.

Park, H.-S., Lee, S.-W., 1996. Off-line recognition of large set hand-written characters with multiple hidden Markov models. Pattern Recognition 29, 231–244.

Pavlidis, T., Zhou, J., 1992. Page segmentation and classification. Comput. Vision Graphics Image Process. 54, 434–496.

Postl, W., 1986. Detection of linear oblique structure and skew in digitized documents. In: Proc. Eighth Internat. Conf. on Pattern Recognition. pp. 464–468.

Senior, A.W., Robinson, A.J., 1998. An off-line cursive handwritten recognition system. IEEE Trans. PAMI 20, 309–321.

Srihari, S.N., Govindraju, V., 1989. Analysis of textual images using the hough transform. Machine Vision Appl. 2, 141–153.

Uchida, S., Taira, E., Sakoe, H., 2001. Nonuniform slant correction using dynamic programming. ICDAR, 434–438.

Yan, H., 1993. Skew correction of document images using interline cross-correlation. CVGIP: Graphical Models Image Process. 55, 538–543.