

To appear in Pattern Recognition

A Robust And Fast Skew Detection Algorithm for Generic Documents

Bin Yu and Anil K. Jain

Department of Computer Science, Michigan State University

East Lansing, MI 48824-1027

binyu@cps.msu.edu, jain@cps.msu.edu

Abstract

A robust and fast skew detection algorithm based on hierarchical Hough transform is proposed. It is capable of detecting the skew angle for various document images, including technical articles, postal labels, handwritten text, forms, drawings and bar codes. The algorithm is robust even when black-margins introduced by photocopying are present in the image and when the document is scanned at a low resolution of 50 dpi. The algorithm consists of two steps. In the first step, we quickly extract the centroids of connected components using a graph data structure. Then, a hierarchical Hough transform (at two different angular resolutions) is applied to the selected centroids. The skew angle corresponds to the location of the highest peak in the Hough space. The performance of the algorithm is shown on a number of document images collected from various application domains. The algorithm is not very sensitive to algorithmic parameters. For an A4 size document image scanned at 50 dpi (typically 413×575 pixels), our algorithm is able to detect the skew angle with an accuracy of 0.1° in 0.4 seconds of CPU time on a Sun Sparc 20 workstation.

Keywords: skew detection, document image processing, hierarchical Hough transform, block adjacent graph, connected components.

1 Introduction

A text line in a document is defined as a group of characters, symbols, and words that are adjacent, relatively close to each other, and through which a straight line can be drawn. The

dominant orientation shared by major text lines in a document determines the skew angle of that document. A document with a zero skew angle should have horizontally or vertically printed text lines which are parallel to the respective edges of the paper. Figures 1(a) and (b) show synthetic document images with zero skew angle and a skew angle of 45° , respectively. Nonzero skew may be introduced when a document is scanned or photocopied. Many of the important document analysis algorithms, including optical character recognition (OCR) and page layout analysis, are sensitive to the orientation (or skew) of the input document image. With the rapid growth of the document entry and interpretation systems, it is important to develop algorithms to perform skew detection and correction automatically. In this paper we are concerned with situations where the skew in the whole document image is due to a rigid rotation of the document (Fig. 1(b)). The more general problem of skew detection for document in which the text is not aligned along straight lines (Fig. 1(c)) is beyond the scope of this paper. Some papers deal with situations where there is a nonuniform skew in an image [1].

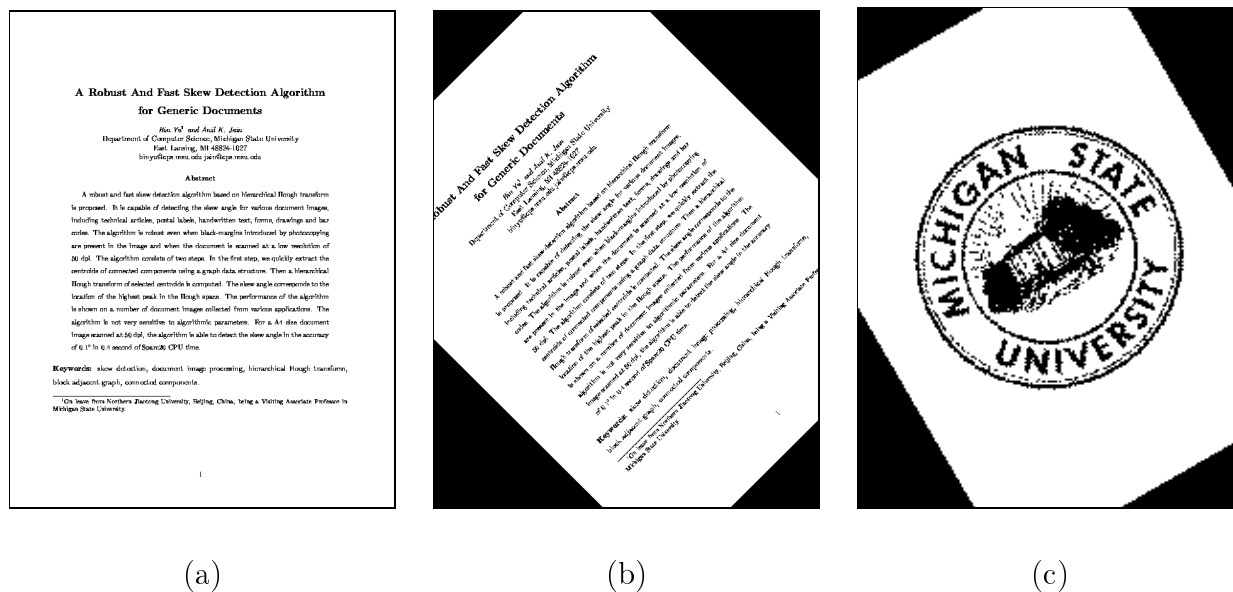


Figure 1: A synthetic document image with (a) zero skew angle, (b) a skew angle of 45° , (c) a document image where text is not aligned along a straight line.

A generic document of interest in this paper can be a machine printed or handwritten text on paper or envelope, a page of technical article composed of text lines of different fonts, bar codes, drawings, figures and tables, a form and a photocopied document with

black margins and noise.

Several attempts to detect the skew angle in a document image have been reported. Most of the popular methods use projection profile (PP) [2] and its variants [3, 4, 5, 6]. These methods usually work well in the case of text only. Hough transform (HT) has also been used for skew detection [7, 8, 9] which can achieve a high detection accuracy. The common weakness of these two approaches is that their computational complexity is proportional to the desired detection accuracy and range. Therefore, most PP or HT-based methods limit their detection ranges, typically to within $(-15^\circ, 15^\circ)$. Another approach uses k - NN (nearest neighbor) clustering [10, 11] of the connected components. This approach has a relatively high accuracy but a large computational cost [12], independent of the detection range, $\mathcal{O}(N^2)$, where N is the number of connected components.

The major computational cost in skew detection methods primarily involves (i) locating the text lines and (ii) skew angle estimation using PP, k - NN clustering or HT. Approaches based on morphological transforms [13], text baseline extraction [14], connected component extraction [3, 9] and burst image creation from black run lengths [8] have been proposed to improve the efficiency of locating text lines. In any event, the algorithmic complexity is a key feature characterizing this preprocessing algorithm in document analysis.

Most previous skew detection methods work well for documents containing printed text, but fail in situations where the text lines are not in majority. Without recognizing character and/or understanding the document, humans usually determine document orientation by collecting some regularly aligned symbols, mostly text. A symbol line is defined as a group of regularly aligned symbols that are adjacent, relatively close to each other, and through which a straight line can be drawn. The symbols can be individual machine printed or handwritten characters with various fonts and sizes, small groups of attached characters (which can be due to low scanning resolution), stripes in a bar code, layout token, etc. For example, each symbol is a connected component in our algorithm.

The skew detection algorithm presented in this paper consists of two steps. The first step quickly extracts the centroids of connected components based on a graph data structure, called block adjacency graph (BAG) [15] and the selection of components based on the size of their bounding boxes. The second step is the skew detection by a hierarchical Hough

transform (HHT) algorithm which performs the skew detection at two angular resolution levels, coarse and fine, with an improved HT. A similar approach was proposed by Rondel and Burel [16] who designed two neural networks for coarse and fine detection. A block diagram of the proposed algorithm is shown in Fig. 2. The BAG can be created during scanning since it is a one-pass algorithm. Based on this graph structure, connected components in document image can be computed more efficiently than traditional methods. On the other hand, the HHT proposed in this paper typically takes only 7% of the computational cost of the basic Hough transform (BHT).

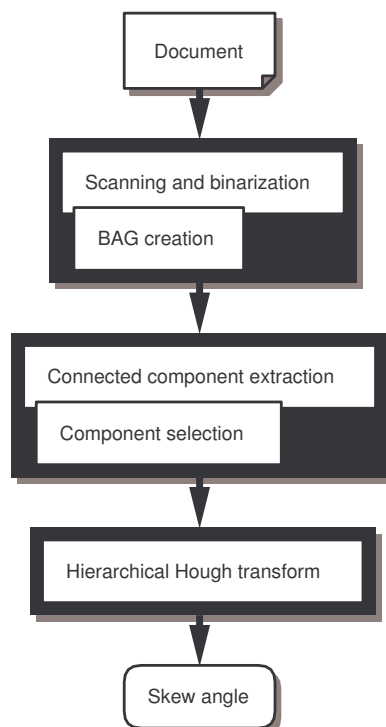


Figure 2: Block diagram of the skew detection algorithm.

The proposed algorithm has been evaluated on a large number of document images, including technical articles from journals, text on address labels on envelopes and postal cards, handwritten text, bar codes and forms. The algorithm is relatively fast and robust for detecting the skew in a generic document.

We will discuss the extraction and selection of connected components in Section 2 and explain the HHT in Section 3. The improvements in HT is described in Section 4. In Section 5 a number of test images and the detected skew angles are given to show the performance

of the proposed algorithm. We make concluding remarks in Section 6.

2 Extracting Centroids of Connected Components

The information needed to determine the document skew is one or more symbol lines, each consisting of several separately aligned symbols. If a group of symbols is closely aligned along a line then their centroids are also roughly aligned along a line with the same direction. The centroid, therefore, is regarded as the salient feature for skew detection.

Yu et al. [15] described the block adjacency graph (BAG) which was used to extract the contours of the connected components in a binary image. This data structure can be created concurrently when the document is being scanned row by row. Processing steps such as contour extraction, vectorization and feature extraction are efficiently implemented based on this data structure [17].

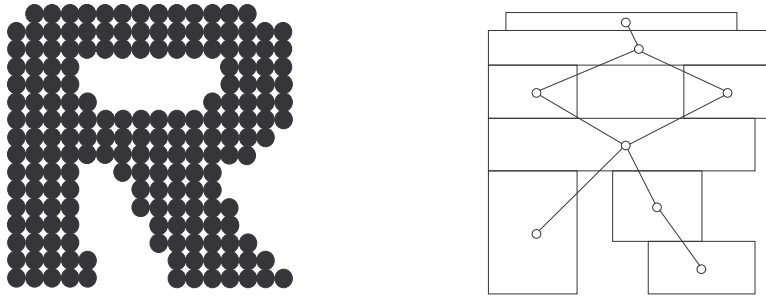


Figure 3: A connected component and its BAG.

Figure 3 shows a connected component and its BAG which consists of eight blocks connected. Note that each block is an approximate bounding box of a partial area in the component and each area involves either one or a series of runs connected and closely justified in margin, given a tolerance. The algorithm for creating BAG is given in Fig. 4.

It is easy to obtain the centroid coordinates of a connected component by searching the graph and computing the weighted average of the centroids of the blocks connected in the BAG. Since HT is relatively time consuming, instead of pixels or runs, we apply it to the set of centroids of connected components, which greatly reduces the computational cost. Since symbol boundaries are actually thick lines of sparse density, using centroids of connected components instead of using pixels or runs as features will also increase the accuracy of skew

```

Each run length in the first row of the input image is regarded as a block.
For the successive rows in the image {
  For each run length  $\mathcal{R}_c$  in current row {
    If  $\mathcal{R}_c$  is 8-connected to a run length in the preceding row {
      If  $\mathcal{R}_c$  is 8-connected to only one run length  $\mathcal{R}_l$  and the horizontal
      positions of their beginning and ending pixels are, respectively, within a
      given tolerance  $\mathcal{T}$ , then  $\mathcal{R}_c$  is merged into the block involving  $\mathcal{R}_l$ .
      Else,  $\mathcal{R}_c$  is regarded as a new block, initialized with links to those
      blocks which are 8-connected to  $\mathcal{R}_c$ .
    }
  }
  Else,  $\mathcal{R}_c$  is regarded as a new block, initialized with no links.
}
}

```

Figure 4: Algorithm for generating BAG data structure.

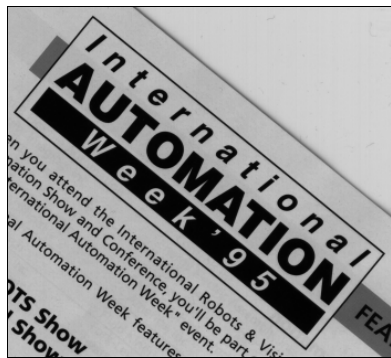
detection. Furthermore, some connected components are filtered out with respect to the size of their bounding boxes, which are introduced due to noise, black margins, lines in tables or forms, figures, etc. Therefore, only useful data are fed to the HT. Figure 5 shows the intermediate results of our algorithm. Here, Fig. 5(a) shows an input document with skew, 5(b) is the corresponding binary image, 5(c) is the BAG data structure, 5(d) contains the connected components extracted based on the BAG, 5(e) shows selected connected components and their centroids which are fed to the HT, and 5(f) shows the deskewed image.

3 Skew Detection by Hierarchical Hough Transform

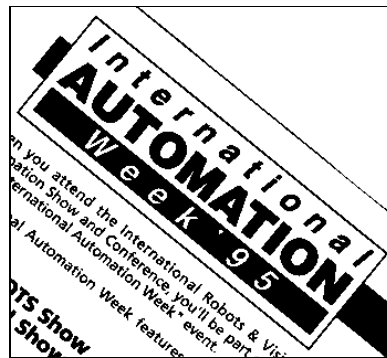
Hough Transform [18] is a well known technique for detecting lines and curves in an image. For line detection, a set of points in Cartesian space $(x - y)$ are mapped to sinusoidal curves in the Hough space $(\rho - \theta)$ via the following transform:

$$\rho = x \cos \theta + y \sin \theta. \quad (1)$$

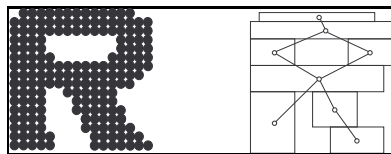
Each time a sinusoidal curve intersects another curve at a particular value of ρ and θ , the likelihood that a line corresponding to that (ρ, θ) coordinate value is present in the original image is increased by using an accumulator array $\mathcal{H}(\rho, \theta)$. The computational steps in the



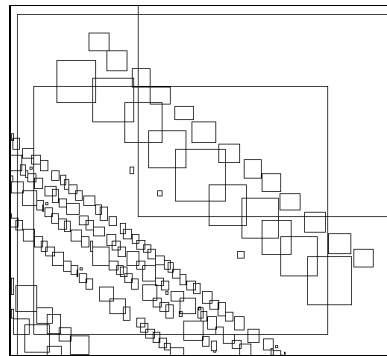
(a)



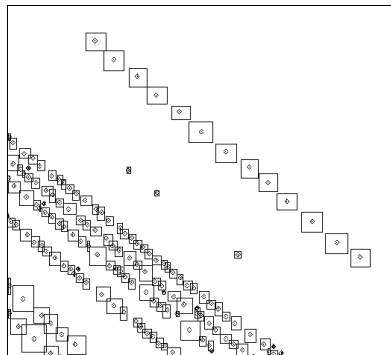
(b)



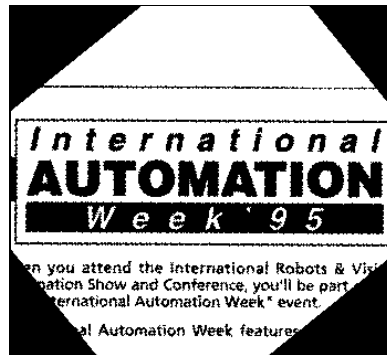
(c)



(d)



(e)



(f)

Figure 5: Intermediate results in the skew detection algorithm: (a) a grey scale document image with skew; (b) binary image of (a); (c) BAG; (d) connected components; (e) selected components and their centroids which are fed to the HT; (f) the skew-corrected image after skew detection.

For all points (x, y) { For $\theta_1 \leq \theta < \theta_2$ { Calculate $\rho = x \cos \theta + y \sin \theta$. Increment $\mathcal{H}(\rho, \theta)$. } } } Find the maximum in the array $\mathcal{H}(\rho, \theta)$.

Figure 6: Computational steps in basic Hough transform.

BHT are given in Fig. 6.

The computational cost of the BHT is $\mathcal{O}_c(N\Psi)$, where N is the number of points in the Cartesian space and Ψ is the number of different values of θ used in constructing the accumulator array. Note that $\Psi = \Delta\theta/\delta\theta$, where $\Delta\theta = \theta_2 - \theta_1$, $0^\circ \leq \theta_1, \theta_2 \leq 180^\circ$ and $\theta_1 < \theta_2$. The parameter $\delta\theta$ is the angular resolution which controls the accuracy of the skew detection (orientation of the line). For computational reasons, most skew detection algorithms using the HT limit themselves to only a subset of $(0^\circ, 180^\circ)$ [12, 8].

This computational cost can be decreased by reducing the number of points (x, y) and $\Delta\theta$ or increasing the interval $\delta\theta$. Initially, we can select a relatively large value of $\delta\theta$, say, $\delta\theta_1$ to approximately determine the skew angle $\hat{\theta}$ in the desired angular range $\Delta\theta$. Then, we can implement the BHT again with the desired angular resolution $\delta\theta_0$ within the reduced range $(\hat{\theta} - \delta\theta_1, \hat{\theta} + \delta\theta_1)$, $\delta\theta_0 < \delta\theta_1$. We call this method the hierarchical Hough transform (HHT). The computational complexity of HHT is $\mathcal{O}_c(N\Psi_1) + \mathcal{O}_c(N\Psi_2) = \mathcal{O}_c(N\Psi_H)$, where

$$\Psi_H = \Psi_1 + \Psi_2 = \frac{2\delta\theta_1}{\delta\theta_0} + \frac{\Delta\theta}{\delta\theta_1}. \quad (2)$$

To minimize the value of Ψ_H , we take the derivative of Eq. (2),

$$\frac{d\Psi_H}{d\delta\theta_1} = 0.$$

Solving for $\delta\theta_1$, we get

$$\delta\theta_1 = \sqrt{0.5\Delta\theta\delta\theta_0} \quad (3)$$

and

$$\Psi_H = \sqrt{\frac{8\Delta\theta}{\delta\theta_0}}.$$

For traditional HT,

$$\Psi_T = \frac{\Delta\theta}{\delta\theta_0}.$$

Therefore,

$$\frac{\Psi_H}{\Psi_T} = \sqrt{\frac{8\delta\theta_0}{\Delta\theta}}.$$

Equation (3) helps us determine the optimum value of $\delta\theta_1$ for given values of $\Delta\theta$ and $\delta\theta_0$. Typically, $\Delta\theta = 180^\circ$ and $0.05^\circ \leq \delta\theta_0 \leq 0.5^\circ$. If $\delta\theta_0 = 0.1^\circ$, then from Eq.(3) we have $\delta\theta_1 = 3^\circ$ and $\Psi_H/\Psi_T = 0.07$ which reduces the complexity of the HT from $\mathcal{O}_c(1800N)$ to $\mathcal{O}_c(120N)$. At the same time, the storage requirement which is proportional to Ψ is reduced from $\mathcal{O}_s(\Psi_T)$ to $\mathcal{O}_s(0.5\Psi_H)$.

4 Improvements in Hough Transform

To match the requirement of the hierarchical strategy for skew detection, the traditional HT as given in Fig. 6 is improved in two respects. In the HT given in Fig. 6, the value assigned to a cell in the Hough space is the number of the sinusoid curves going through this cell. In order to avoid the possibility that the desired skew angle is not in the neighborhood of the angle $\hat{\theta}$ obtained during coarse detection, i.e. $(\hat{\theta} - \delta\theta_1, \hat{\theta} + \delta\theta_1)$, we use an integral of the sinusoid curve within a cell in the Hough space as the value assigned to it. For a simple example involving a two-point HT (see Fig. 7), the improved HT algorithm will find the correct peak at 144° rather than at 108° . Note that both the accumulator cells are assigned a value of 2 by the basic HT algorithm.

To compensate for the quantization error, we can use a formal error model during the increment step [19, 20]. Another solution to this problem is to replace the accumulator value by an average of the neighborhood values. These two methods have been shown to be equivalent under the assumption of isotropic errors [21]. On the other hand, the symbol lines with similar angles but different polar distances should have the same contribution in skew detection. A two-dimensional window can be introduced for this purpose where all the

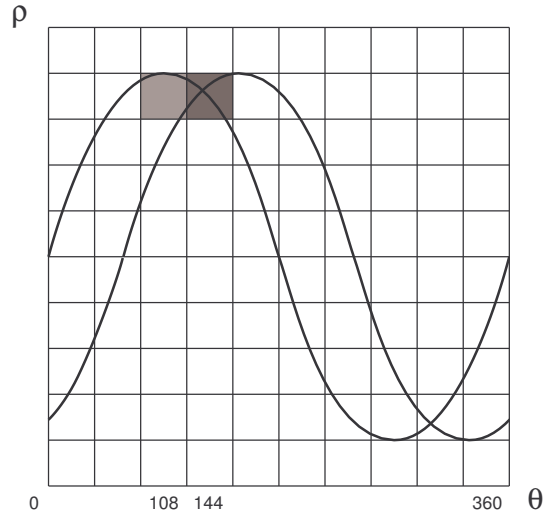


Figure 7: The improved HT algorithm will find the correct peak at 144° .

```

Define  $\rho(\theta) = x \cos \theta + y \sin \theta$ .
For all points  $(x, y)$  {
  For  $\theta$  in  $(\theta_1, \theta_2)$  {
    For  $\rho$  in  $(\rho(\theta), \rho(\theta + \delta\theta))$  {
      Increment  $\mathcal{H}(\rho, \theta)$  by the curve integral of  $\rho(\theta)$  within the
      cell  $(\theta, \theta + \delta\theta) \times (\rho, \rho + \delta\rho)$ .
    }
  }
}
Find the maximum in the array  $\mathcal{H}(\rho, \theta)$  in a 2D window.

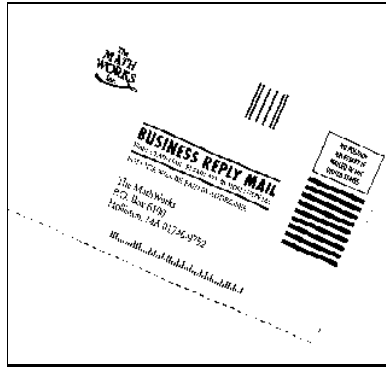
```

Figure 8: The improved basic Hough transform algorithm.

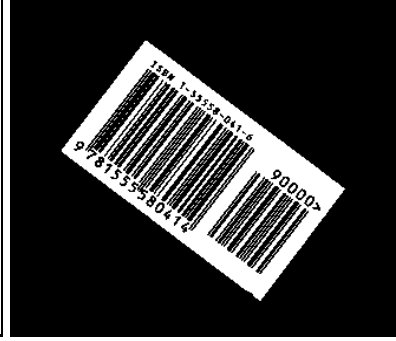
accumulator cells inside the window with values larger than a threshold value will be summed. The skew angle is the weighted average within that window where the peak is found. The weight values are based on the entries in the accumulator array. Based on this process, the detected peak is more robust in situations where the document image has several different candidate directions due to a non-rigid skew or text/symbol lines with different orientations. The improved HT algorithm is given in Fig. 8.



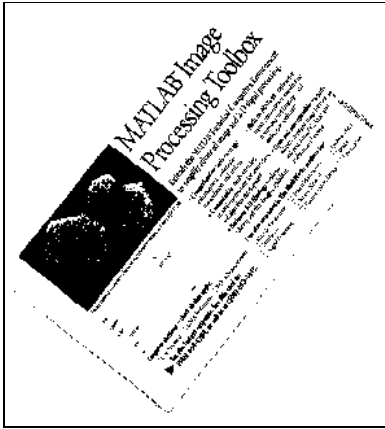
(a)



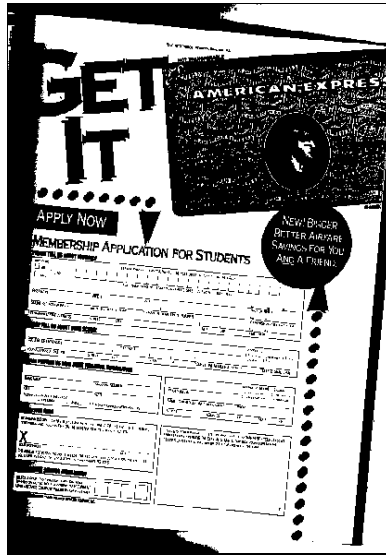
(b)



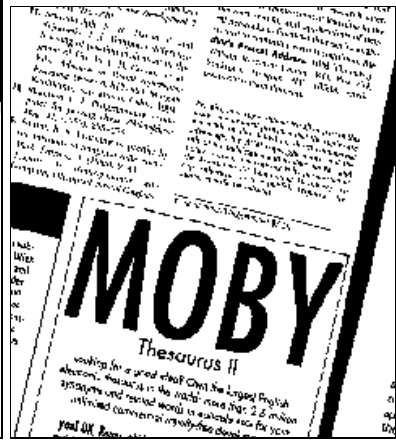
(c)



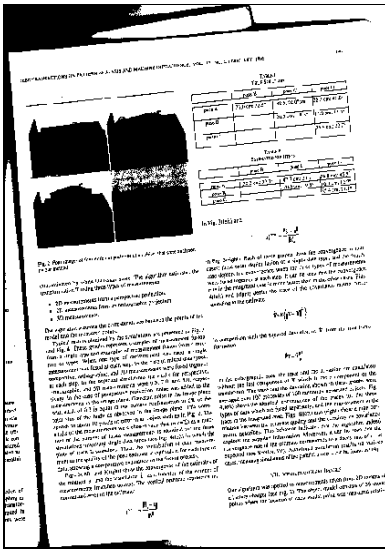
(d)



(e)



(f)



(g)



(h)



(i)

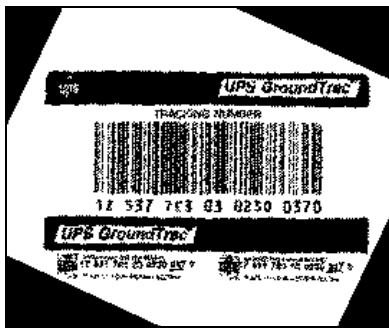
Figure 9: Input document images for skew detection: (a) a postal label; (b) a post card; (c) a bar code; (d) a coupon; (e) an application form in yellow pages; (f) photocopied document; (g) a page in the IEEE TPAMI; (h) and (i) advertisements in magazines.

Table 1: Performance of skew detection.

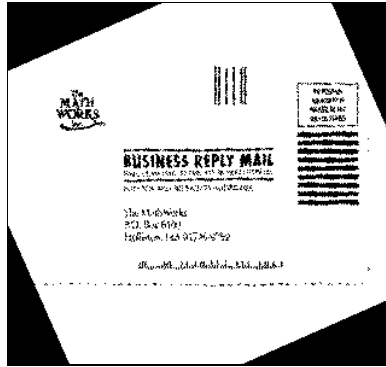
Image (Fig. 9)	Size (pixels)	Resol. (dpi)	Time (s)		Estimated skew angle (degrees)
			HHT	BHT	
(a)	268×225	50	0.1	0.9	25.1
(b)	319×300	50	0.1	3.6	-23.8
(c)	326×286	100	0.1	0.5	-36.9
(d)	297×327	50	0.1	4.7	51.2
(e)	415×596	50	0.4	13.5	-5.1
(f)	238×268	50	0.1	14.7	-13.5
(g)	422×598	50	0.5	34.8	4.5
(h)	414×587	50	0.4	15.3	3.1
(i)	412×576	50	0.2	6.3	-4.8

5 Experimental Results

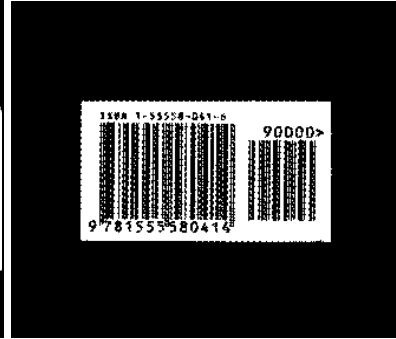
The skew detection algorithm proposed in this paper has been applied to a large number of document images to test its capability in different applications. These images were obtained by scanning envelopes, journals, magazines, postal labels, yellow pages and coupons using an HP ScanJet IICx 24-bit color scanner. Some of the scanned images are shown in Fig. 9. The image shown in Fig. 9(a) is scanned from a postal label, in 9(b) from a post card, in 9(c) from the back cover of a handbook, in 9(d) from a coupon, in 9(e) from a yellow page, in 9(f) from a magazine, in 9(g) from the IEEE Trans. on PAMI, and in 9(h) and 9(i) from an advertisement brochure. Note that thick black margins appear in the images shown in Figs. 9(c), (e), (f), (g) and (h). The images in Fig. 9 contain a variety of fonts, logos, tables and bar codes. Table 1 gives information about the input image sizes, scanning resolution and estimated skew angle. The corresponding processing times of our algorithm and the traditional BHT algorithm on a SPARC20 CPU are also shown in Table 1. In our experiments, we use $\delta\rho = 4$, $\delta\theta_0 = 0.1^\circ$, $\Delta\theta = 180^\circ$ and, therefore, $\delta\theta_1 = 3^\circ$. The algorithm works well on all of these document images which have different characteristics and resolutions. The correct orientations of the images based on the estimated skew angle are shown in Fig. 10. However, our algorithm failed to detect the correct skew for some of these images when the scanning resolution was reduced further (for example, Fig. 9(c) at 50 dpi and Fig. 9(f) at 20 dpi).



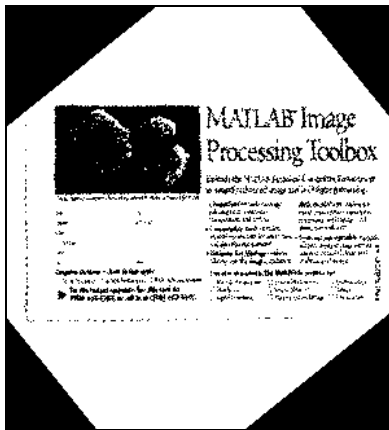
(a)



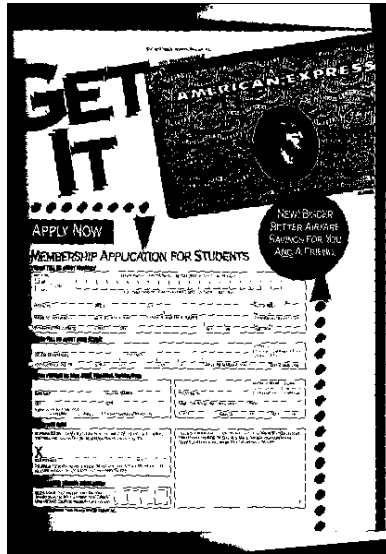
(b)



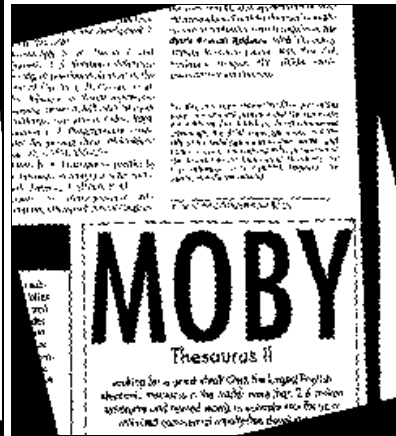
(c)



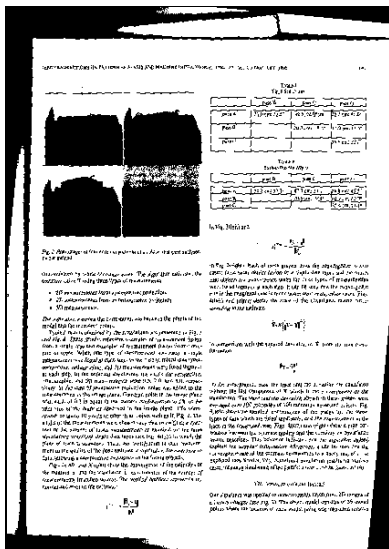
(d)



(e)



(f)



(g)



(h)



(i)

Figure 10: Skew corrected results for the images shown in Fig. 9.

The skew detection accuracies reported in the literature are difficult to compare because they depend on many factors such as the accuracy of the scanner and the scanning resolution. Our algorithm was also tested on synthetic document images of 16 pages of this manuscript at 75 dpi (the scanned image of the first page is shown in Fig. 1). By synthetic we mean that these images were not scanned and the skew was introduced manually. Our results indicate that we can obtain an accuracy of 0.1 degree, except for the pages containing Figs. 5, 9 and 10. The HHT method shows its power especially when $\delta\theta_0 < 0.1$, but it is not suitable in the case of $\delta\theta_0 > 1$ (when the value of Ψ is not very large), because of the large cell size in the Hough space.

6 Conclusions

The HT can be used to accurately detect the document skew. However, its computational complexity is a limiting factor, especially when a high detection accuracy and a wide detection range are required. The existing methods using the HT are usually able to detect the skew angle in a limited angular range to avoid its huge computation and storage requirements. Compared with the traditional skew detection methods using the HT, our algorithm has the following three significant features: (i) instead of pixels and runs, we use centroids which are quickly extracted based on the BAG data structure, (ii) the HT is implemented hierarchically with different angular resolutions and within different angular ranges, and (iii) the HT algorithm is improved in cell value calculation and peak detection. These features allow our algorithm to quickly detect the skew angle of relatively more complex document images scanned or subsampled at a low resolution with a high accuracy and within the whole angular range ($0^\circ, 180^\circ$). The algorithm has been shown to be able to handle document images scanned at a resolution of 50 dpi and images which contain noise, black margins, figures, and forms.

We currently use a commercial scanning software (PixelFx) and, therefore, the BAG creation is performed only after the entire scanning of the document is completed. More than half of the total processing time needed by our algorithm is used to create the BAG. If the BAG creation module can be inserted inside the scanning and binarization module (see

Fig. 2), then this time-consuming procedure can be completed during the scanning itself and the proposed skew detection algorithm could operate in “real-time”.

It is worth mentioning that usually only a small part of a document is sufficient for skew detection. This will greatly reduce the processing time. However, quickly determining the significant area of the document for skew detection is an open problem.

Acknowledgment

We would like to thank Mr. Sharath Pankanti for providing the source code for the BHT and Miss Yu Zhong for her assistance in preparing this manuscript.

References

- [1] C. L. Yu, Y. Y. Tang, and C. Y. Suen, “Document skew detection based on the fractal and least squares method,” in *Proc. of the 3rd Int. Conf. on Document Analysis and Recognition*, (Montreal), pp. 1149–1152, 1995.
- [2] W. Postl, “Detection of linear oblique structures and skew scan in digitized documents,” in *Proc. of the 6th Int. Conf. on Pattern Recognition*, (Paris), pp. 687–689, 1986.
- [3] H. S. Baird, “The skew angle of printed documents,” in *Proc. of SPIE 40th Annual Conf. and Sym. on Hybrid Imaging Systems*, pp. 21–24, 1987.
- [4] T. Akiyama and N. Hagita, “Automated entry system for printed documents,” *Pattern Recognition*, vol. 23, pp. 1141–1154, 1990.
- [5] T. Pavlidis and J. Zhou, “Page segmentation by white streams,” in *Proc. of the 1st Int. Conf. on Document Analysis and Recognition*, (Paris), pp. 945–953, 1991.
- [6] Y. Ishitani, “Document skew detection based on local region complexity,” in *Proc. of the 2nd Int. Conf. on Pattern Document Analysis and Recognition*, (Tskuba Science City), pp. 49–52, 1993.

- [7] S. N. Srihari and V. Govindaraju, “Analysis of textual images using the Hough transform,” *Machine Vision and Applications*, vol. 2, pp. 141–153, 1989.
- [8] S. Hinds, J. Fisher, and D. D’Amato, “A document skew detection method using run-length encoding and the Hough transform,” in *Proc. of the 10th Int. Conf. on Pattern Recognition*, (Atlantic City), pp. 464–468, 1990.
- [9] Y. Nakano, Y. Shima, H. Fujisawa, J. Higashino, and M. Fujinawaet, “An algorithm for the skew normalization of document image,” in *Proc. of the 10th Int. Conf. on Pattern Recognition*, (Atlantic City), pp. 8–13, 1990.
- [10] A. Hashizume, P. S. Yeh, and A. Rosenfeld, “A method of detecting the orientation of aligned components,” *Pattern Recognition Letters*, vol. 4, pp. 125–132, 1986.
- [11] L. O’Gorman, “The document spectrum for structural page layout analysis,” *IEEE Trans. Pattern Anal. and Machine Intell.*, vol. 15, pp. 1162–1173, 1993.
- [12] L. O’Gorman and R. Kasturi, Eds., *Document Image Analysis*. IEEE Computer Society Press, 1995.
- [13] S. Chen and R. M. Haralick, “An automatic algorithm for text skew estimation in document images using recursive morphological transforms,” in *Proc. of the First Int. Conf. on Image Processing*, (Austin), pp. 139–143, 1994.
- [14] J. Liu, C. M. Lee, and R. B. Shu, “An efficient method for the skew normalization of a document image,” in *Proc. of the 11th Int. Conf. on Pattern Recognition*, (The Hague), pp. 122–125, 1992.
- [15] B. Yu, X. Lin, Y. Wu, and B. Yuan, “Isothetic polygon representation for contours,” *CVGIP: Image Understanding*, vol. 56, pp. 264–268, 1992.
- [16] N. Rondel and G. Burel, “Cooperation of multi-layer perceptrons for the estimation of skew angle in text document images,” in *Proc. of the 3rd Int. Conf. on Document Analysis and Recognition*, (Montreal), pp. 1141–1144, 1995.

- [17] B. Yu, “A method for automatic understanding of symbol connected diagrams,” in *Proc. of the 3rd Int. Conf. on Document Analysis and Recognition*, (Montreal), pp. 803–806, 1995.
- [18] J. Illingworth and J. Kittler, “A survey of the Hough transform,” *Computer Graphics and Image Processing*, vol. 44, pp. 87–116, 1988.
- [19] S. D. Shapiro, “Properties of transforms for the detection of curves in noisy pictures,” *Comput. Vision Graphics Image Process.*, vol. 8, pp. 219–236, 1978.
- [20] S. D. Shapiro, “Feature space transforms for curve detection,” *Pattern Recognition*, vol. 10, pp. 129–143, 1978.
- [21] D. H. Ballard, “Generalizing the Hough transform to detect arbitrary shapes,” *Pattern Recognition*, vol. 13, pp. 111–122, 1981.