

中央研究院
資訊科學研究所

Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-05-011

Estimation of Skew Angles for Scanned Documents Based on Piecewise Covering by Parallelograms

Fu Chang, Chien-Hsing Chou, and Shih-Yu Chu



September 2005 || Technical Report No. TR-IIS-05-011

<http://www.iis.sinica.edu.tw/LIB/TechReport/tr2005/tr05.html>

Estimation of Skew Angles for Scanned Documents Based on Piecewise Covering by Parallelograms

Fu Chang, Chien-Hsing Chou, and Shih-Yu Chu

Institute of Information Science, Academia Sinica, Taipei, Taiwan, R.O.C.

Email: fchang@iis.sinica.edu.tw, ister@iis.sinica.edu.tw, sarasate@iis.sinica.edu.tw

Corresponding Author:

Fu Chang

Telephone: 886-02-2788-3799 ext. 1819

Fax: 886-02-2782-4814

Email: fchang@iis.sinica.edu.tw

Postal Address: Institute of Information Science, Academia Sinica, 128, Academia Road,
Section 2, Taipei 115, Taiwan.

Abstract. We propose a fast and robust skew estimation method for scanned documents that estimates skew angles based on piecewise covering of objects, such as textlines, figures or tables. The method first divides a document image into a number of non-overlapping slabs in which each object is covered by parallelograms. It then estimates the skew angle based on these parallelograms or, equivalently, their complementary regions. Putting this method to a systematic test and comparing it with a few alternatives, we find favorable results for our method in terms of accuracy rates, sensitivity to non-textual objects, effectiveness in dealing with documents of unspecified reading order, and computational efficiency. Some work is also conducted to find an effective way to further shorten its computation time at the expense of an extremely small loss of accuracy.

Keywords: Document image analysis, piecewise covering, skew angle, skew angle estimation

1 INTRODUCTION

Many document image analysis methods, such as those for page layout analysis and for optical character recognition, are sensitive to the orientation of document images. To cope with this problem, the skew angle of a document image must be estimated and the orientation of the image adjusted accordingly. This paper addresses the estimation of skew angles for scanned documents that result from the placement of an original document at an improper angle relative to a certain axis. Several methods have been proposed to deal with this issue [1-19]. An overview of these works can be found in articles [20-21].

In the above works, projection-based (PJ) methods are the most commonly used [1-4]. They form projection profiles of a document image at various angles and calculate the skew angle of the document from the achieved maximum value. Some methods calculate skew angles using the nearest neighbor (NN) clustering technique [5-8]. In these methods, the angle between each connected component and its nearest neighbor is computed, and a histogram of these angles is formed to determine the skew angle. Some other approaches apply Hough transform (HT) to find skew angles by selecting the peak value in the Hough plane [8-12]. There is also the method that finds skew angles based on the maximum variance of transition counts (TC) [13-14], and methods that base their angle estimations on cross-correlations (CC) [15-19]. Most of the above methods are designed for documents comprised of horizontal textlines only. However, the methods in [14-15, 18] are designed to handle documents of mixed reading orders. When dealing with these documents, it applies the same operations twice: once on the original image, and once on a 90^0 rotation of the image.

The NN and HT methods have relatively higher computational costs. The complexity of the NN method is in the order of C^2 , where C is the number of components. When applied to Chinese or Japanese documents, NN is further complicated by the need to search for com-

patible neighbors because Chinese or Kanji characters are a two-dimensional juxtaposition of components. The complexity of the HT method, on the other hand, is in the order of PR , where P is the number of pixels and R is the range of line orientations. In addition to its costly time consumption, HT requires a large memory space to store intermediate data in the Hough plane. This method, moreover, is sensitive to the occurrence of non-textual objects in the documents. The other three methods, i.e., PJ, TC, and CC have lower computational costs and impose fewer constraints on their applications. They are thus good candidates for comparison with our method.

In this paper, we propose a method that uses piecewise covering of objects by parallelograms. Although it can deal with all types of objects, it handles rectangular objects particularly well, including groups of aligned characters, textlines, tables, rectangular pictures, and field separators. In our approach, all objects are covered by parallelograms constructed at various angles. The better the covering, the smaller the area occupied by the covering parallelograms, or, equivalently, the larger the area occupied by the complementary regions. The orientation of the document is then inferred to be the angle at which the complementary regions occupy the largest area.

Of the three alternative methods that are compared with our approach, the PJ and CC methods take advantage of the fact that documents are composed of textlines and white spaces between them. The two methods examine different features that reflect the above fact. The TC method takes advantage of the fact that passing through textlines gives rise to higher transition counts than passing through straight lines of arbitrary orientations. Our method, on the other hand, takes advantage of the fact that most textual or non-textual objects in documents are rectangular in shape. It therefore attempts to cover them with parallelograms. We deliberately cover these objects in a piecewise manner to limit, as much as possible, the nega-

tive influence of one type of object on the construction of parallelograms for another type of object.

Since fast computation is an objective of skew angle estimation, none of the above methods, including ours, explicitly model the objects from which they derive the estimates for skew angles. Thus, the effectiveness of these methods has to be assessed by experiments, rather than by analysis or intuitive arguments. One of the aims of this paper is to provide a systematic evaluation of the above-mentioned methods. The following aspects of these methods are evaluated: the effects of non-textual objects on their performance accuracy, their effectiveness in dealing with documents with unspecified reading orders, and their computational efficiency. The evaluation results are expressed in terms of accuracy rates.

The remainder of this paper is organized as follows. In Section 2, the procedure of skew angle estimation is introduced. In Section 3, the experiment results showing the performance of our method are given. In Section 4, we discuss the performance issues of our method, its sensitivity to certain conditions, and a way to speed up skew angle computation. Finally, in Section 5, we present our conclusion.

2 SKEW ANGLE ESTIMATION

We first vertically divide a document image into a number of non-overlapping regions. As shown in Fig.1a, the image is divided into two regions, called *slabs*. We set the width of each slab at 450 pixels for images produced at a resolution of 300dpi. For a document whose width is not a multiple of 450 pixels, the last slab will have a width smaller than 450 pixels. The sensitivity of our method to different slab widths is given in Section 4, which shows that our method is not sensitive to this particular choice of width. To determine the orientation of a given image, we process it at various angles and pick the one that is most appropriate for the

image's orientation. To analyze the image from all possible angles, we draw parallel scan lines at these angles.

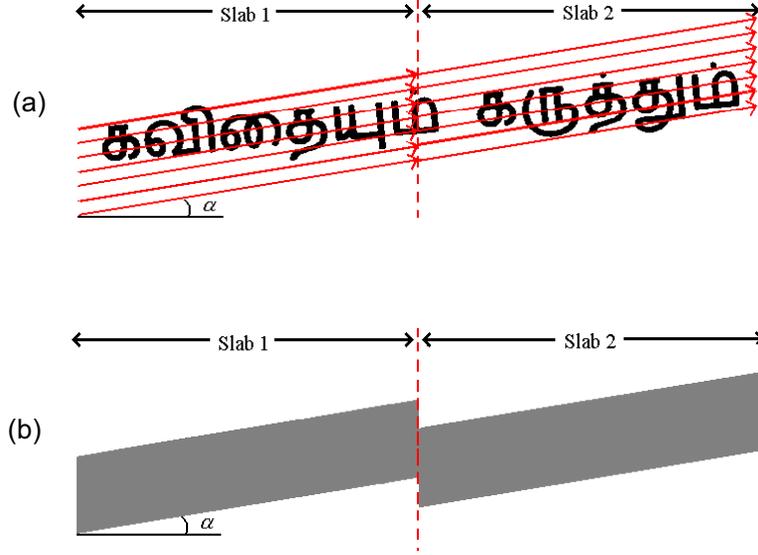


Fig. 1. (a) Parallel scan lines with a skew angle α . (b) Parallelograms constructed by changing certain sections of the scan lines into a gray color.

In each slab, we examine whether there are any black pixels on a scan line. If there is at least one black pixel on a section of the scan line, we change the color of all the pixels on that section to gray. Otherwise, the pixels stay white. By so doing, we cover all line objects in a slab with gray parallelograms (Fig. 1b), between which are white regions; that is, regions without any black pixels. We further illustrate this in Figures 2a through 2c. The document image in Fig. 2a is covered by parallelograms constructed at skew angle -4.6° and skew angle 2° , shown in Figures 2b and 2c, respectively. Since the orientation of the image is -4.6° , it is clear that the parallelograms constructed at this angle result in larger white regions than those constructed at 2° .

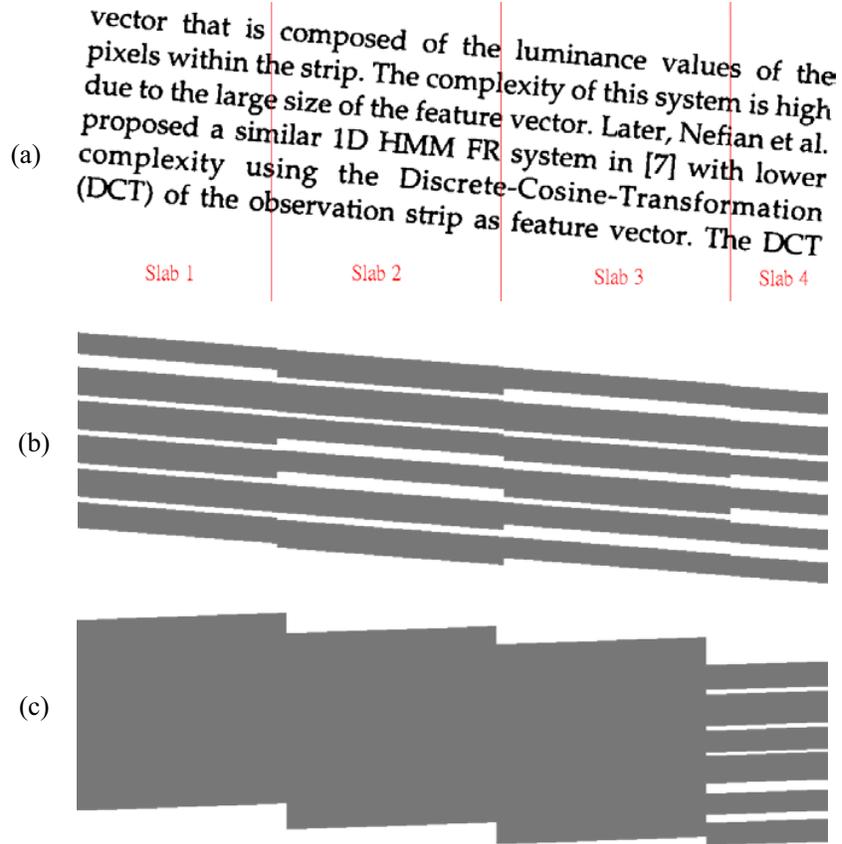


Fig. 2. (a) A document consisting of horizontal textlines, rotated at -4.6° . (b) The image approximated by parallelograms constructed at a -4.6° skew angle. (c) The image approximated from a 2° skew angle.

In the final step, let $A(\theta)$ be the area of the parallelograms constructed at skew angle θ . Let T be the total area of the document image and $B(\theta) = T - A(\theta)$. Then, $B(\theta)$ is the area of the complementary regions of the parallelograms. The orientation of the document is set at $\theta^* = \underset{\theta}{\text{argmax}} B(\theta)$.

To calculate the size of a white region, we simply count the number of white scan lines. White scan line computation is performed efficiently by the following steps.

IF we detect a black pixel on the scan line
THEN we skip the rest of the scan line and move to the next one
ELSE we increase the count of white scan lines by one

(1)

In the above operation, we assume that the skew angle of each document image is within the range $[-15^\circ, 15^\circ]$. Our method first searches for the best skew angle A , using a 2° skew angle as the search step size (i.e., we construct parallelograms at $-14^\circ, -12^\circ, \dots, 12^\circ, \text{ and } 14^\circ$), after which it selects the best skew angle B from the following three angles: $A-1^\circ, A$, and $A+1^\circ$. Finally, it searches for the best skew angle C within the range $(B-1^\circ, B+1^\circ)$, using a 0.1° skew angle as the search step size. Angle C is taken as the orientation of the test image.

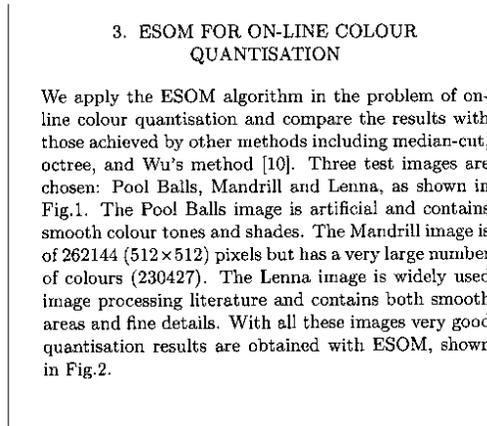


Fig. 3. A text region with a vertical separator to its left.

We now present some examples to illustrate the performance of our method. In the first example, we demonstrate that the division of an image into slabs helps to obtain accurate estimates of the skew angle. As shown in Fig. 3, there is a vertical separator that is taller than the text region on its right. If we do not divide the image into slabs or, equivalently, if we set the slab size as the image size, the parallelograms constructed at all angles will have the same number of pixels because they all have the same height and the same base (see Fig. 4a and Fig. 4b as examples). This means that we cannot derive a unique estimate, let alone an appropriate one, from a covering constructed in this way. If, however, we set the slab size at the usual 450 pixels, we can restrict the dominating effect of the separator in the first slab. In this

way, the parallelograms constructed at 0^0 (Fig. 4c) create more white spaces in the second and third slabs than those constructed at, for example, 6^0 (Fig. 4d).

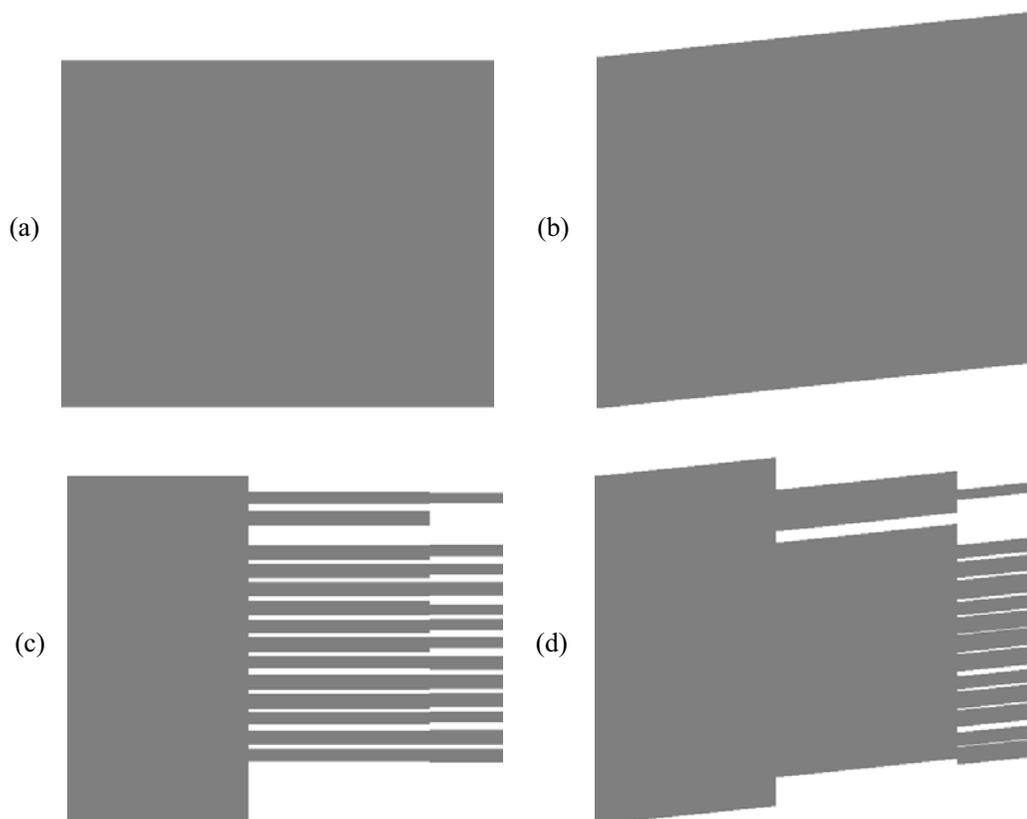


Fig. 4. The coverings constructed at angle (a) 0^0 , and (b) 6^0 , when the slab size is set as the image size. The coverings constructed at angle (c) 0^0 , and (d) 6^0 , when the slab size is set at 450 pixels.

Our second example relates to the reading order of textlines. Chinese and Japanese documents contain horizontal textlines, vertical textlines, or a mixture of both. When the reading order is not specified in advance, the skew angle is usually estimated by applying the same operations twice; once to the original image and once to the 90^0 rotation of the image [14-15, 18]. One benefit of our method is that it does not have to perform this double operation because it treats all documents in the same way that it treats English documents. Fig. 5a shows a handwritten Chinese document comprised of four vertical textlines. The results of constructing coverings of this image at skew angles 7^0 and 2^0 are shown in Fig. 5b and Fig. 5c, respec-

tively. The orientation of this document image is 7° , which our method detects correctly. The covering constructed at this angle creates clear white regions that are not apparent in the covering constructed at angle 2° . Conceptually, the objects covered in this case are not vertical textlines, but groups of horizontally aligned characters. Our method does not rely on a full alignment of characters horizontally; partial alignment is sufficient.

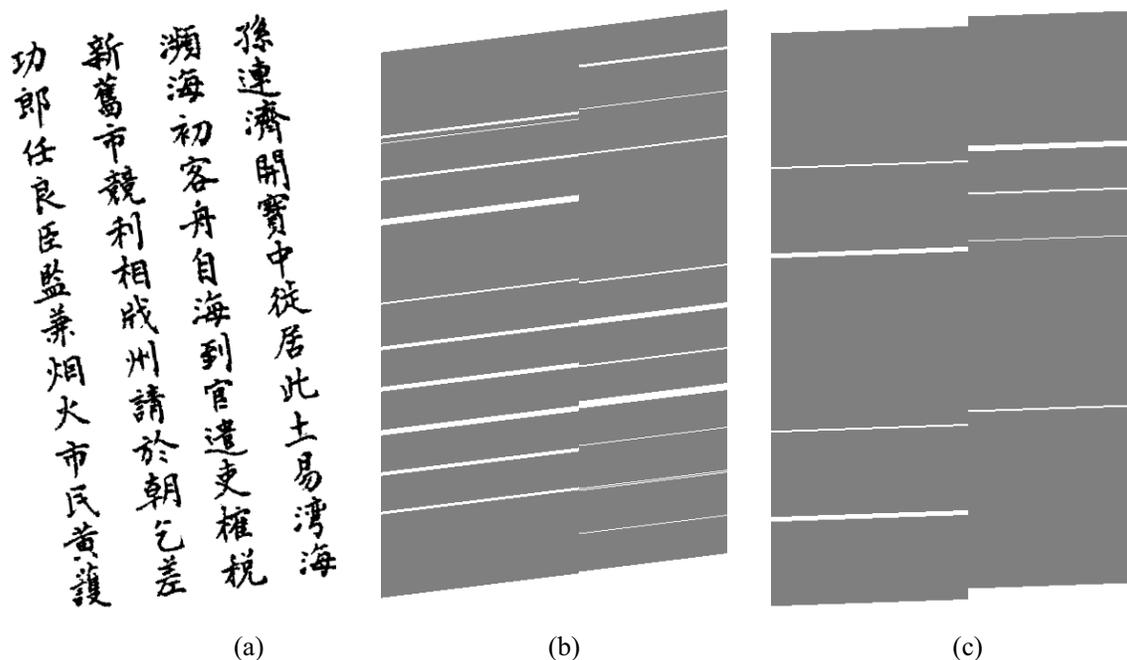
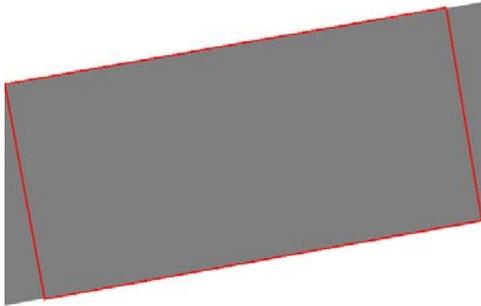


Fig. 5. (a) A document consisting of vertical textlines, rotated at 7° . (b) Parallelograms constructed at skew angle 7° . (c) Parallelograms constructed at skew angle 2° .

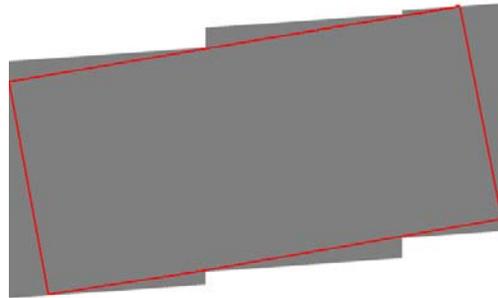
In fact, our method is suitable for documents of either reading order or of mixed reading orders, so long as the textlines in the documents form rectangular blocks. The rectangular blocks do *not* even have to be text regions. Our approach can also deal with a rectangular picture. Fig. 6a shows a document consisting of a single picture rotated at 9.8° . The results of constructing a covering of this image at angles of 9.8° and 4° are shown in Fig. 6b and Fig. 6c, respectively. Our method detects the largest white regions in the approximation constructed at 9.8° . Since most figures in documents are rectangular, our method can handle these documents rather easily.



(a)



(b)



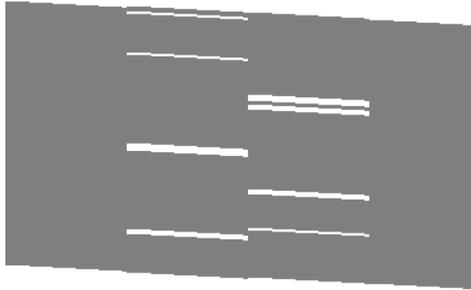
(c)

Fig. 6. (a) A document consisting of only one picture, rotated at 9.8° . (b) Parallelograms constructed at skew angle 9.8° . (c) Parallelograms constructed at skew angle 4° .

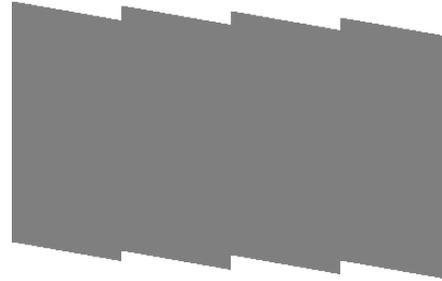
A similar situation is found in documents consisting of tables. Fig. 7a shows a document comprised of a filled-in table, rotated at -3° , while Fig. 7b and Fig. 7c show the covering constructed at -3° and -10° , respectively. Our method detects the orientation of the table image at -3° . Since tables are always rectangular in shape, they are valuable sources of information for our method to find the orientation of the documents that contain them.

姓名: 王乃堅	性別: 男
E-Mail: wang@henry.ee.ntust.edu.tw	
服務機構: 國立台灣科技大學	
單位: 電機系	職稱: 副教授
通訊地址: 台北市基隆路4段439	
聯絡電話: (02) 737-6708	傳真: (02) 737-6699

(a)



(b)



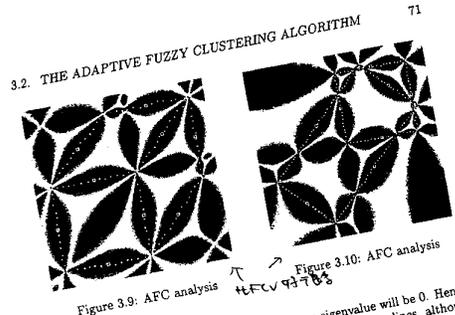
(c)

Fig. 7. (a) Test image consisting of a filled-in table, rotated at -3° . (b) Parallelograms constructed at skew angle -3° . (c) Parallelograms constructed at skew angle -10° .

Our last examples, shown in Fig. 8a and Fig. 8b, contain both textlines and figures. Fig. 8a has vertical textlines, while Fig. 8b has horizontal textlines. The skew angles of these two documents are set at -8.1° and 10.7° , respectively, and our method estimates them at -8.2° and 10.6° , respectively. The coverings constructed at the estimated skew angles are shown in Fig. 8c and Fig. 8d. These two examples demonstrate that our method works well with a combination of textual and non-textual elements.

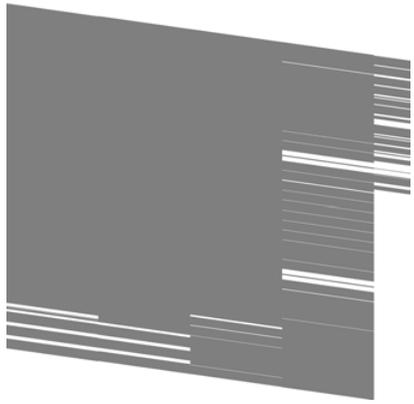


(a)

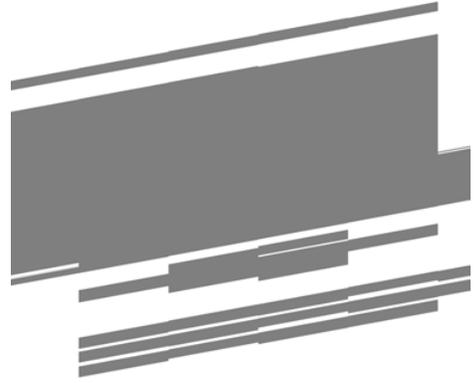


expansion vertical to this plane is 0. Thus, one eigenvalue will be 0. Hence, α becomes 1, and the algorithm searches for ideal straight lines, although the cluster may have a circular expansion within the plane.

(b)



(c)



(d)

Fig. 8. (a) Vertical textlines combined with a large figure, rotated at -8.1° . (b) Horizontal textlines that are mixed with a large figure, rotated at 10.7° . (c) Parallelograms constructed at -8.2° , which is our estimated angle for (a). (d) Parallelograms constructed at 10.6° , which is our estimated angle for (b).

3 EXPERIMENT RESULTS

To evaluate the proposed method, we collected a variety of document images from newspapers, books, magazines, and journals. The images include several types of objects, such as horizontal textlines, vertical textlines, tables, and pictures. All documents, or parts of documents, are scanned at 300dpi and pasted onto an A4-sized blank canvas, resulting in $2,371 \times 3,151$ pixels in each image. The images are further rotated at various angles to pro-

vide skewed documents; the maximum possible angle is restricted to $\pm 15^\circ$. A total of 500 images are produced in this manner, and divided into the following five categories (also shown in Table 1). 1) English documents with no figures or tables (Fig. 9). 2) Documents in traditional/simplified Chinese or Japanese (Fig. 10). 3) Documents comprised of horizontal textlines and large-scale figures (Fig. 11). 4) Documents comprised of textlines and tabular regions (Fig. 12). 5) Documents in several languages, including Greek, Arabic, Hindi, Nepalese, Tibetan, Persian, Bulgarian, Thai, Vietnamese, and Hebrew. All of these images are in horizontal textlines, and some contain figures (Fig. 13).

Table 1. Test samples.

	Document Type	Number of Test Images
1 st Category	English Documents	100
2 nd Category	Chinese and Japanese Documents	100
3 rd Category	Documents Containing Large-Scale Figures	100
4 th Category	Documents Containing Tables	100
5 th Category	Multilingual Documents	100



Fig. 9. Examples of the first category of documents.

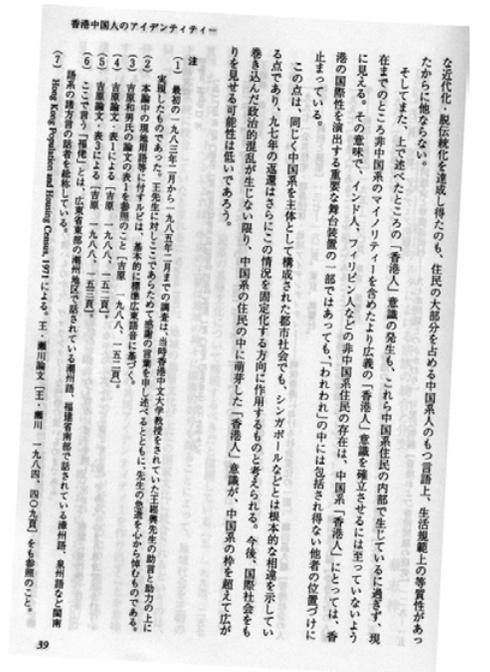


Fig. 10. Examples of the second category of documents.

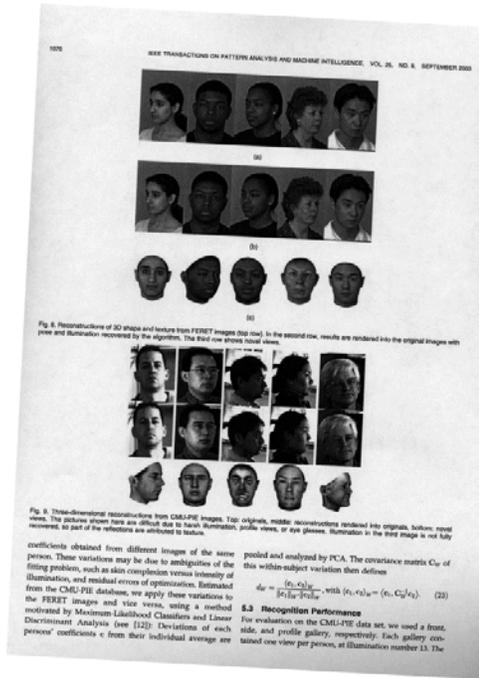


Fig. 8. Reconstructive of 3D shape and texture from FERET images (top row). In the second row, results are rendered into the original images with the illumination recovered by the algorithm. The third row shows facial views.

Fig. 9. Three-dimensional reconstructions from CMU-PIE images. Top: original, middle: nonoverlapping rendered into original, bottom: rendered views. The pictures shown here are difficult due to each illumination, profile views, or eye glasses. Illumination in the first stage is not fully recovered, so part of the reflections are attributed to texture.

coefficients obtained from different faces of the same person. These variations may be due to ambiguities of the fitting problem, such as skin complexion versus intensity of illumination, and residual errors of optimization. Extracted the FERET images and view using the minimum variance method, using a method motivated by Maximum Likelihood Classifiers and Linear Discriminant Analysis (see [12]). Deviations of each person's coefficients c from their individual average are pooled and analyzed by PCA. The covariance matrix C_{ij} of this within-subject variation then defines

$$d_{ij} = \frac{(c_i - \bar{c}_i)(c_j - \bar{c}_j)}{\sqrt{(\sigma_i^2 - \bar{\sigma}_i^2)(\sigma_j^2 - \bar{\sigma}_j^2)}} \text{ with } (c_i, \bar{c}_i)_{i=1}^n = (c_{i1}, C_{ij} c_{i2}) \quad (22)$$

3.3 Recognition Performance
For evaluation on the CMU-PIE data set, we used a front, side, and profile gallery, respectively. Each gallery contained one view per person, an illustrative number is 13. The

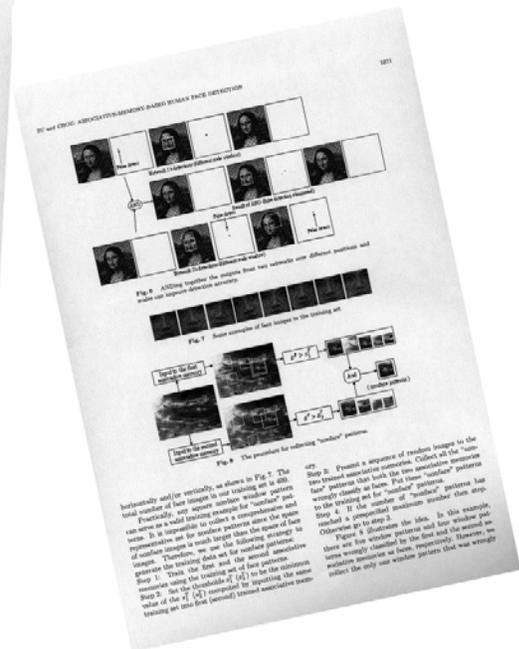


Fig. 9. Three-dimensional reconstructions from CMU-PIE images. Top: original, middle: nonoverlapping rendered into original, bottom: rendered views. The pictures shown here are difficult due to each illumination, profile views, or eye glasses. Illumination in the first stage is not fully recovered, so part of the reflections are attributed to texture.

Fig. 10. A sequence of random images to the two-class associative memory. Collect all the "non-face" patterns that both the two associative memories were trained on as faces. Put these "non-face" patterns to the training set for "non-face" patterns. Set n to the number of "non-face" patterns. Step 4: In this example, we use $n=10$ to train a proposed associative memory.

Fig. 11. A sequence of random images to the two-class associative memory. Collect all the "non-face" patterns that both the two associative memories were trained on as faces. Put these "non-face" patterns to the training set for "non-face" patterns. Set n to the number of "non-face" patterns. Step 4: In this example, we use $n=10$ to train a proposed associative memory.

Fig. 11. Examples of the third category of documents.

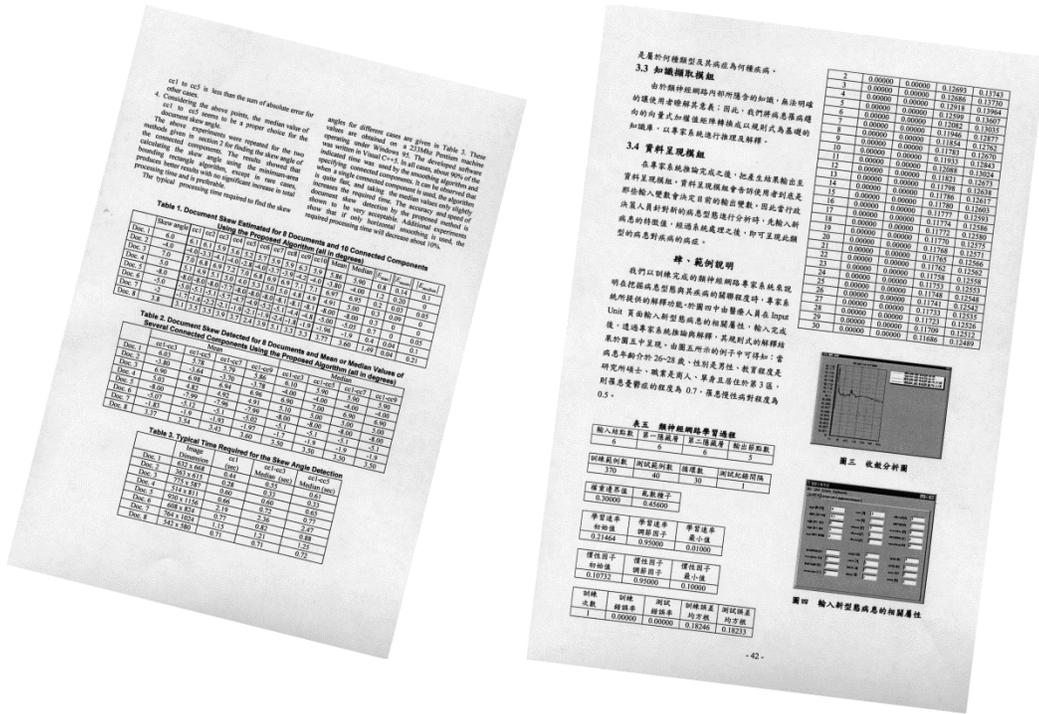


Fig. 12. Examples of the fourth category of documents.

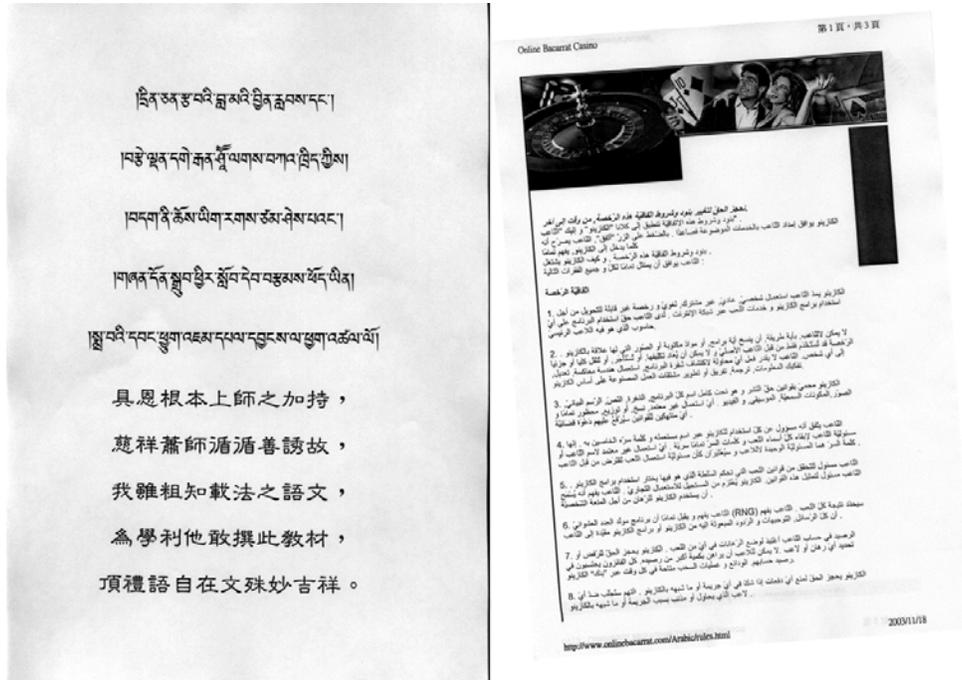


Fig. 13. Examples of the fifth category of documents.

For comparison, we apply the PJ, TC, and CC methods to the same set of test images. For the PJ method, we follow the paper of Postl [1], which projects an image along various angles. The angle it picks as the skew angle of the document is the one at which the difference between the maximal and the minimal projected value is the largest. For TC, we implement the method proposed by Chen and Wang [14], which has also been proposed independently by Ishitani [13]. Chen and Wang extend the application of this method to documents with mixed reading orders. For CC, we implement the method of Avanindra and Chaudhuri [17], which applies a metric proposed by Yan [16]. Instead of applying the metric to the whole image as Yan does, Avanindra and Chaudhuri apply it to fixed-sized blocks and choose the median value among these blocks as a representative of the image. As claimed by the authors, and confirmed by our experiments, this has the merit of resisting the negative effects of pictorial areas on the skew angle estimation. It is also worth mentioning that [1] (PJ method) and [14] (TC method) require no parameter values to be set offline, while [17] (CC method) has two

parameters (separation parameter d and block size) whose values must be fixed at the outset. We adopt the parameter values suggested in [17].

Except for test images in the 2nd category, we apply all methods to each image once only; that is, we apply them horizontally. For the 2nd-category images, derived from Chinese and Japanese documents, we obtain two sets of results from all methods, except ours. One set is obtained by applying the methods once, and the other set by applying them twice. The results are compared with ours that are obtained horizontally.

There are two kinds of estimation error. One consists of the average and variance of errors of all test images, while the other consists of the average and variance of errors of selected test images. In the latter case, the images with the worst 20% error rates for a given method are dropped from the calculation of the average rate for that method. This type of error rate is used to exclude the possible influence of worst-case performances. The experiment results show that our method achieves the best results of all the methods compared in the experiment, irrespective of which type of errors rates are involved.

Recall that our method is conducted in such a way that the step size for searching for the optimal skew angle is first set at 2^0 , then 1^0 , and then 0.1^0 . This technique shortens the time required to find the optimal angle without affecting the accuracy. For this reason, we use the same technique in our implementations of the PJ and TC methods. The CC method, on the other hand, has its own search technique [17], which we also adopt in our implementation.

The results for the 1st category are listed in Table 2. This category consists of images derived from English documents that have either a single column or two columns. All methods perform rather well. Note that in this table, All Images contains the results obtained from all test images, while Top 80% contains the results obtained from the top 80% of test images.

The best results are expressed in boldface figures. The same format is adopted in subsequent tables.

Table 2. Error rates ($^{\circ}$) of the four methods for the 1st-category images.

1 st Category		Our Method	PJ	TC	CC
Average	All Images	0.149	0.23	0.185	0.166
	Top 80%	0.102	0.153	0.148	0.115
Variance	All Images	0.129	0.206	0.18	0.144
	Top 80%	0.096	0.14	0.131	0.109

The results for the 2nd category, which consists of images derived from Chinese and Japanese documents, are listed in Table 3. Among the images, 75 of them contain textlines of a single reading order that are either horizontal or vertical, while another 25 have textlines of mixed reading orders. When the three alternative methods are applied horizontally, their results are listed under Once, and when they are applied twice, the results are listed under Twice. The results show that the three alternative methods have inferior performances under Once, whereas our method performs rather well, even though it is applied to the images only once.

Table 3. Error rates ($^{\circ}$) of the four methods for the 2nd-category images.

2 nd Category		Our Method	PJ Once	PJ Twice	TC Once	TC Twice	CC Once	CC Twice
Average	All Images	0.139	0.644	0.496	0.238	0.171	0.409	0.18
	Top 80%	0.088	0.304	0.254	0.173	0.108	0.209	0.132
Variance	All Images	0.143	0.896	0.591	0.202	0.155	0.297	0.192
	Top 80%	0.07	0.296	0.263	0.16	0.091	0.122	0.096

From the results in Tables 2 and 3, we conclude that our method performs almost as well on English documents as it does on Chinese and Japanese documents, even though most Chinese and Japanese characters are square in shape and, in addition, English characters have

more variations in height due to the distinctions between upper and lower cases, and between ascenders and descenders.

The results for the 3rd category are shown in Table 4. This category contains images derived from English documents with proportionally large figures, some of which are even non-rectangular in shape. To a certain extent, all methods are affected by the prevalence of figures in these documents, as can be seen from the increased error rates in this category. Among them, PJ is most affected. In one test image, the ground-truth angle is 12° , but PJ estimates it at -13° . CC is less affected, while TC and our method are the least affected.

Table 4. Error rates ($^\circ$) of the four methods for the 3rd-category images.

3 rd Category		Our Method	PJ	TC	CC
Average	All Images	0.231	7.787	0.249	0.345
	Top 80%	0.178	3.419	0.183	0.223
Variance	All Images	0.135	9.049	0.223	0.325
	Top 80%	0.102	4.934	0.144	0.186

The results for the 4th category are shown in Table 5. This category is made up of images derived from English documents that contain tables, all of which are rectangular in shape. The numbers in them are horizontally aligned. It is not surprising, therefore, that all methods perform rather well for this category.

Table 5. Error rates ($^\circ$) of the four methods for the 4th-category images.

4 th Category		Our Method	PJ	TC	CC
Average	All Images	0.111	0.16	0.15	0.139
	Top 80%	0.062	0.096	0.078	0.075
Variance	All Images	0.127	0.163	0.18	0.146
	Top 80%	0.073	0.105	0.084	0.078

The results for the 5th category, which contains images derived from multilingual documents, are shown in Table 6. Some documents may contain figures, most of which are rectangular in shape. All textlines in the documents are horizontal. In this category, PJ is the only method that performs relatively poorly, compared to other methods.

Table 6. Error rates ($^{\circ}$) of the four methods for the 5th-category images.

5 th Category		Our Method	PJ	TC	CC
Average	All Images	0.077	2.05	0.176	0.197
	Top 80%	0.051	0.208	0.105	0.129
Variance	All Images	0.075	5.816	0.24	0.23
	Top 80%	0.05	0.264	0.072	0.125

To summarize, in Fig. 14 we plot the average errors of the four methods over All Images and over Top 80% images. The error rates of the three alternative methods for the 2nd-category images are obtained from applying them twice to the images. The average computation times required for the four methods are shown in Table 7. The processing times listed in the table result from applying the four methods to the *entire* document images in a test environment comprised of an Intel Pentium 4 CPU 2.2GHz with a 256 MB RAM. There are two sets of times in this table. The first set results from applying the four methods to all test images once; the second results from applying our method to the 2nd-category images once, and applying the alternative methods to the same images twice. From these results, we conclude that our method achieves the smallest averages and variances of errors, while consuming the least amount of computation time. In terms of the average error rate, TC ranks second, CC ranks third, and PJ is the last. In terms of the time consumption, PJ, TC and CC run second, third, and fourth, respectively.

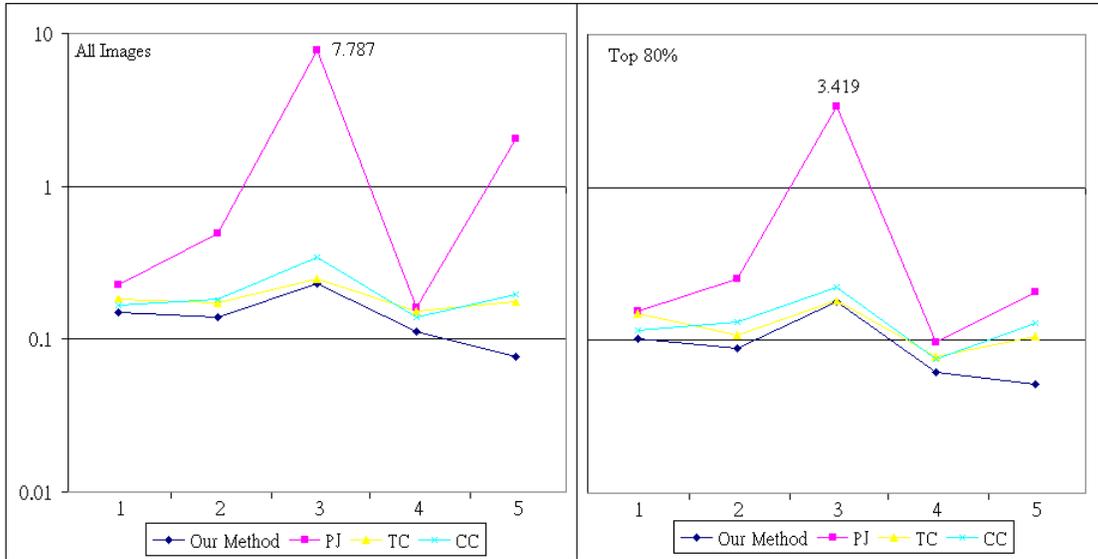


Fig. 14. Left: Average errors over all images. Right: Average errors over top-80%images.

Table 7. Average computation time (s) of the four methods.

	Our Method	PJ	TC	CC
All images	1.53 (Once)	2.03 (Once)	3.95 (Once)	10.58 (Once)
2 nd -Category Images	1.66 (Once)	4.04 (Twice)	7.8 (Twice)	21.47 (Twice)

4 DISCUSSION

Here, we discuss the performance issues of our method and, in particular, its sensitivity to certain conditions, such as slab widths, proportions of black pixels, etc. We also propose an effective way to speed up the computation of skew angles. The limitations of our method are also described at the end of this section.

Slab Widths. Through our experiments with different slab widths, we have produced the average error rates of various slab widths. Although the results, listed in Table 8, show that setting the slab width at 450 produces the best average outcome over all documents, they also suggest that setting the slab width at any value between 150 and 750 produces a relatively

satisfactory performance. This fact shows that our method is *not* overly sensitive to different slab widths.

Table 8. Average estimation errors ($^{\circ}$) with different slab widths.

Slab Widths	<u>150</u>	<u>300</u>	<u>450</u>	<u>600</u>	<u>750</u>
1 st Category	0.147	0.152	0.149	0.154	0.162
2 nd Category	0.135	0.138	0.139	0.143	0.142
3 rd Category	0.224	0.222	0.231	0.228	0.231
4 th Category	0.116	0.113	0.111	0.111	0.124
5 th Category	0.091	0.083	0.077	0.078	0.082
All Documents	0.143	0.142	0.141	0.142	0.148

Processing Time. In the logical step (1) in Section 2, the process continues to examine each pixel on a scan line until it encounters a black pixel. Since this constitutes the major step of our method, the processing time is negatively correlated with the proportion of black pixels in an image. Table 9 lists the average processing times for different proportions of black pixels in an image. In the worst case, when the proportion is 0% (i.e., the document is blank), our method requires 2.83 seconds to complete the process. This is, of course, an extremely rare case. In our test database, the proportions of black pixel images range from 9% to 65%, with processing times of 1.83 and 1.13 seconds, respectively. The average proportion of black pixels is 13.5%, with a corresponding processing time of 1.53 seconds.

Table 9. Time required for different proportions of black pixels.

Proportion of Black Pixels (%)	0	9	11	13.5	15	29	48	65
Average Run Time (s)	2.83	1.83	1.55	1.53	1.50	1.38	1.25	1.13

Speedup. Instead of processing the original image, we can work on a low-resolution version of it. As shown in Table 10, the average time to estimate the skew angles of A4-size

documents scanned at 300dpi is 1.53 seconds, while the average time to estimate the angles on the 100dpi version of the same images is only 0.26 seconds, including the time to convert 300dpi images into 100dpi images. Processing of the compressed version of an image, moreover, gives rise to comparable performances. Table 11 lists the average error rates of our method in the five categories at a 100dpi resolution, where the slab width is set at 150. Table 12 compares the average error rates of the four different methods over all images at the two different resolutions. The averages of the three alternative methods are taken from the results of applying them twice to the 2nd-category images and the results of applying them once to all other images. Note that our method achieves a better average at 100dpi than the other methods achieve at 300dpi.

Table 10. Average computation time of our method for two different image resolutions.

	300dpi	100dpi
Time Consumption (s)	1.53	0.26

Table 11. Average error rates ($^{\circ}$) of our method over all 100dpi images.

Category		1	2	3	4	5
Average	All Images	0.152	0.158	0.397	0.115	0.103
	Top 80%	0.109	0.085	0.243	0.051	0.061

Table 12. The average error rates ($^{\circ}$) of the four methods over all images.

Methods		Our Method at 300dpi	Our Method at 100dpi	PJ at 300dpi	TC at 300dpi	CC at 300dpi
Overall	All Images	0.141	0.185	2.115	0.186	0.205
Average	Top 80%	0.096	0.110	0.826	0.124	0.135

Limitations. Our method has two limitations. First, since it constructs covering by filling any slab lines that contain at least one black pixel, it is vulnerable to images that are rich in

noise in the background area. While it is rare to encounter such poor images in a normal situation, they certainly constitute a limitation to our method. One remedy for this potential problem is to apply a pre-process that removes all pepper noises from the images. Since noise removal is desirable for all later processes, such as layout analysis and character recognition, it is certainly a worthwhile step. In our experiments, we passed all images through a noise removal procedure before rotating them to a certain skew angle.

The second limitation relates to the assumption that all textlines on skewed images maintain the same angle relative to a certain axis. This assumption holds if the images are scanned from documents that have parallel textlines. Although most documents have this assumed property, scanning is not the only way to create images from them. Photography is another way to produce images; however, it can destroy textline parallelism in the images. Our method does not work for such images.

5 CONCLUSION

We have presented a robust and fast method of document skew estimation. Without any pre-processing to divide the documents into textual and non-textual areas, the proposed method is applied to the entire document, including all textual, tabular, and pictorial regions. Experiment results prove the effectiveness and robustness of this method. A systematic comparison with other approaches shows that it not only achieves better accuracy than the other methods, but also produces the results in much less time. Speeding up the computation of estimating skew angles is possible by converting an image to a low-resolution one. In so doing, our experiment results show that the speedup factor is large, while the loss of estimation accuracy is extremely small.

REFERENCES

- [1] W. Postl, Detection of linear oblique structures and skew scan in digitized documents, *Proc. 6th Int. Conf. Pattern Recognition*, Paris, pp. 687-689, 1986.
- [2] H. S. Baird, The skew angle of printed documents, *Proc. Conf. Photographic Scientists and Engineers*, vol. 40, pp. 14-21, 1987.
- [3] A. Bagdanov and J. Kanai, Projection Profile Based Skew Estimation Algorithm for JBIG Compressed Images, *Proc. of 4th Inter. Conf. on Document Analysis and Recognition*, pp. 401-405, 1997.
- [4] G. Nicchiotti and C. Scagliola, Generalized projections: a tool for cursive handwriting normalization, *Proc. of 5th Inter. Conf. on Document Analysis and Recognition*, pp. 729-732, 1999.
- [5] X. Jiang, H. Bunke, and D. Widmer-Kljajo, Skew detection of document images by focused nearest-neighbor clustering, *Proc. of 5th Inter. Conf. on Document Analysis and Recognition*, pp. 629-632, 1999.
- [6] 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.
- [7] L. O'Gorman, The document spectrum for page layout analysis, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1162-1173, 1993.
- [8] U. Pal and B. B. Chaudhuri, An improved document skew angle estimation technique, *Pattern Recognition Letters*, vol. 17, no. 8, pp. 899-904, 1996.
- [9] H. F. Jiang, C. C. Han, and K. C. Fan, A fast approach to the detection and correction of skew documents, *Pattern Recognition Letters*, vol. 18, no. 7, pp. 675-686, 1997.

- [10] B. Yu and A. K. Jain, A robust and fast skew detection algorithm for generic documents, *Pattern Recognition*, vol. 29, no. 10, pp. 1599-1729, 1996.
- [11] D. S. Le, G. R. Thoma, and H. Wechsler, Automated page orientation and skew angle detection for binary document images, *Pattern Recognition*, vol. 27, no. 10, pp. 1325-1344, 1994.
- [12] Y. Min, S.-B. Cho, and T. Lee, A data reduction method for efficient document skew estimation based on Hough transformation, *Proc. of 13th International Conference on Pattern Recognition*, pp. 732-736, 1996.
- [13] Y. Ishitani, Document Skew Detection Based on Local Region Complexity, *Proc. of 2nd Inter. Conf. on Document Analysis and Recognition*, pp. 49-52, 1993.
- [14] Y. K. Chen and J. F. Wang, Skew detection and reconstruction based on maximization of variance of transition-counts, *Pattern Recognition*, vol. 33, pp. 195-208, 2000.
- [15] T. Akiyama and N. Hagita, Automated Entry system for printed documents, *Pattern recognition*, vol.23, pp.1141-1154, 1990.
- [16] H. Yan, Skew Correction of Document Images Using Interline Cross-Correlation, *CVGIP: Graphical Models and Image Processing*, vol. 55, no. 6, pp. 538-543, 1993.
- [17] Avanindra and S. Chaudhuri, Robust detection of skew in document images, *IEEE Trans. on Image Processing*, vol. 6, no. 2, pp. 344-349, 1997.
- [18] M. Chen and X. Ding, A robust skew detection algorithm for grayscale document image, *Proc. of 5th Inter. Conf. on Document Analysis and Recognition*, pp. 617-620, 1999.
- [19] B. Gatos, N. Papamarkos, and C. Chamzas, Skew detection and text line position determination in digitized documents, *Pattern Recognition*, vol. 30, no. 9, pp. 1505-1519,

1997.

- [20] J. J. Hull, Document image skew detection: Survey and annotated bibliography, *Document Analysis Systems II*, J. J. Hull and S. L. Taylor (eds), World Scientific, pp. 40-64, 1998.
- [21] R. Cattoni, T. Coianiz, S. Messelodi, and C.M. Modena, Geometric layout analysis techniques for document image understanding: a review, *Technical Report, 9703-09*, ITC-IRST, Trento Italy, 1998.