

Finding the Best-Fit Bounding-Boxes

Bo Yuan¹, Leong Keong Kwoh¹, and Chew Lim Tan²

¹ Centre for Remote Imaging, Sensing and Processing,
National University of Singapore, Singapore 119260
{yuanbo, lkkwoh}@nus.edu.sg

² Department of Computer Science, School of Computing,
National University of Singapore, Singapore 117543
tancl@comp.nus.edu.sg

Abstract. The bounding-box of a geometric shape in 2D is the rectangle with the smallest area in a given orientation (usually upright) that completely contains the shape. The best-fit bounding-box is the smallest bounding-box among all the possible orientations for the same shape. In the context of document image analysis, the shapes can be characters (individual components) or paragraphs (component groups). This paper presents a search algorithm for the best-fit bounding-boxes of the textual component groups, whose shapes are customarily rectangular in almost all languages. One of the applications of the best-fit bounding-boxes is the skew estimation from the text blocks in document images. This approach is capable of multi-skew estimation and location, as well as being able to process documents with sparse text regions. The University of Washington English Document Image Database (UW-I) is used to verify the skew estimation method directly and the proposed best-fit bounding-boxes algorithm indirectly.

1 Introduction

Text blocks in the printed documents are customarily rectangular. In document image analysis, this rectangular contour provides important information about the geometry of the text blocks, such as orientations and dimensions. Normally, text blocks are marked with upright (not best-fit) bounding-boxes as the results of page segmentation processes. However, it is desirable to find the best-fit bounding-boxes of the text blocks so that the intended shape of the blocks can be presented. Finding the best-fit bounding-box of a text block is a minimization problem that searches for the bounding-box with the smallest area among all the possible orientations.

This paper presents an algorithm of finding the best-fit bounding-box of a given text block (textual component group). It can be used to solve certain problems in the field of document image analysis (DIA). Therefore, given a textual document image, the components are first grouped by this grouping function, and then the best-fit bounding-boxes of the individual component groups are detected with the proposed best-fit bounding-box algorithm in this paper.

One of the applications of the proposed best-fit bounding-box algorithm is the skew estimation for textual document images. The skew of a textual document image

is the amount of misalignment of its text lines relative to the edges of the image. Skews are introduced during the digitization process due to the imprecision or difficulty in the placement of the original documents. Skew estimation is one of the important processing steps in document image understanding. There are some in-depth reviews [1]-[3] available for the large array of techniques that have been developed in the research literature [4]-[12]. These skew estimation methods have different detection accuracy, time and space efficiencies, abilities to detect the existence of multiple skews in the same image, and robustness in noisy environments and scan-introduced distortions. The skew estimation method proposed in this paper is capable of detecting and locating multi-skews in a document image, which many of the existing skew detectors are not capable. The full set of the 979 real document images in the University of Washington English Document Image Database (UW-I) is used to evaluate the performances of this skew estimation method. Since there are no publicly available databases with ground truth for directly evaluating of the proposed best-fit bounding-box algorithm, the use of UW-I on skew estimation is an indirect evaluation of the effectiveness of the proposed best-fit bounding-box algorithm.

2 Finding the Best-Fit Bounding-Boxes

The input to the proposed best-fit bounding-box algorithm is a segmented page where the components are properly grouped by *any* page segmentation algorithms. Then, the best-fit bounding-boxes of the individual component groups are detected.

2.1 Components Grouping

We use a simple and efficient component-grouping function that is based on the spatial distances and size similarities among the components [13]. Given two components of areas s_1 and s_2 , if the value of the grouping function $f(s_1, s_2)$ in Eq. (1) is larger than the Euclidian distance between the two, they are considered *directly linked*. A component group is a collection of components among which there always exists at least one path of direct links for any two components. This component-grouping process takes the form of acyclic multi-trees called the *component-linking trees*. The coefficient k in Eq. (1) is given a constant value, but can be adjusted by the batch of samples in use.

$$f(s_1, s_2) = \sqrt{\frac{ks_1s_2}{s_1 + s_2}} \quad (1)$$

The grouping function in Eq. (1) has several desirable properties: (a) it is a distance measure; (b) it is symmetric for any two components; (c) it is rotation invariant; (d) it is resolution invariant, if the aliasing effect is discounted; (e) when the size difference between two components becomes large, the function tends to be determined mainly by the smaller component. Thus, the grouping process biases strongly toward close components with similar sizes, resulting in higher tolerance to the interferences from graphical elements and other source of noises for the textual components.

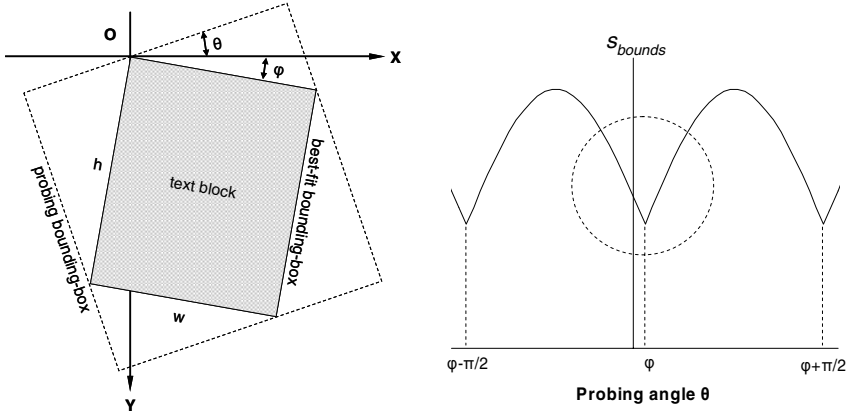


Fig. 1. The configuration for probing the best-fit bounding-box of an ideal text block (left) and the region-of-interest in the area-angle plot (right)

Note that the quality of the component grouping or page segmentation algorithms in use has direct impact on the quality of the best-fit bounding-boxes. The best-fit bounding-boxes are found for whatever blocks that are provided.

It should be pointed out that even though a special component grouping using Eq. (1) is used in this paper, any page segmentation algorithms can be used for the proposed searching algorithm for best-fit bounding-boxes.

2.2 Best-Fit Bounding-Boxes Probing

Given a group of components, its best-fit bounding-box can be found by minimizing the area of its bounding-box. As shown in Fig. 1 (left), the area of a bounding-box S_{bounds} is:

$$\begin{aligned}
 S_{bounds} &= |w \cos(\theta - \phi) + h \sin(\theta - \phi)| |w \sin(\theta - \phi) + h \cos(\theta - \phi)| \\
 &= wh + (w^2 + h^2) |\sin 2(\theta - \phi)| / 2
 \end{aligned}
 \tag{2}$$

Fig. 1 (right) shows the curve of Eq. (2) which has a period of $\pi/2$. In the context of document image analysis, the circled region is the region-of-interest. Generally, the range of page skews is within than $\pm 3^\circ$ for human scan operators. This practical limit can be used to set the bracket for the minimization process.

For an ideal text block, when the probing angle θ is close to the skew angle ϕ of the block, the curve of Eq. (2) approximates a triangle. In real documents, this region may not be a symmetric triangle due to the indent of the first line and/or the shorter last line, which can be observed in Fig. 2.

The minimization process uses the bi-section (successive bracketing) method [14]. Given an initial bracket, such as $[-9^\circ, 9^\circ]$ in Fig. 2, its central value is used to recursively divide the left and the right halves. For any bracket, if the central point is smaller than that of the two end-points, this bracket must contain the minimum, and the other branches of bracketing are abandoned. This process continues until the

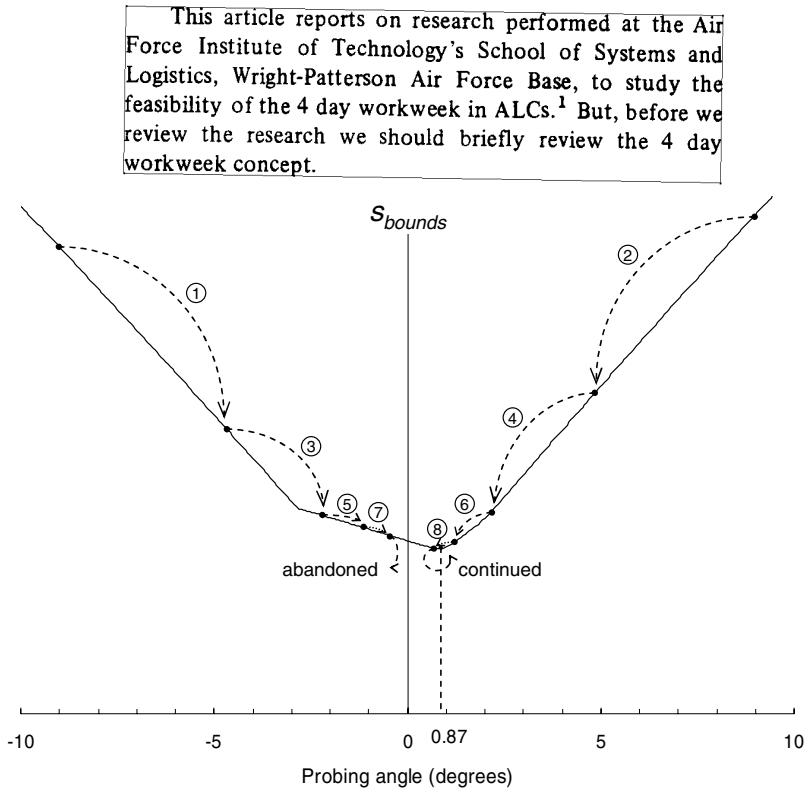


Fig. 2. Bi-section search for the probing bounding-box with the minimum area of a real text block. The curve (bottom) near the minimum is distorted from a symmetric triangular shape due to the indent of the first line and the shorter last line of the text block (top).

width of the current smallest bracket is smaller than, say 0.01° . The middle of this bracket is taken as the orientation of the best-fit bounding-box. The searching range is assumed unimodal.

Even though the proposed best-fit bounding-box algorithm is practically fast enough in its baseline form, there is still room for speedup. One speedup technique is to use the vertices of the convex hulls of the components in the group rather than using the constituting points of the components directly. This alone can achieve an order-of-magnitude reduction in computation time.

To evaluate the efficiency of the proposed best-fit bounding-box algorithm, the full set of the 979 real document images (2592×3300 pixels each) from UW-I are used. Experimental results show that the total time spent on the preprocessing stage (which includes the image input, the connected component analysis, the component filtering and the component grouping) is 1 912 820 milliseconds (or 1.95 seconds per sample), and the total time spent on the proposed best-fit bounding-box algorithm is 40 559 milliseconds (or 0.04 second per sample). Therefore, the computational cost is a non-issue even on legacy computers.

onguchi, K. Hutter, S. B. Savage

n the (x, t) coordinates to the deformation equations

(2.22)

2) are mutually dependent and may arise because μ may depend on the centre of mass motions on variables characterized by centre of mass motion. The degree of this paper via the introduction

bed friction coefficients μ which model type II) or both (models study are numbered from 1 to 8 to their type.

nearly between the front and rear angles held constant. It is also as no variation of μ arises in this ter [1]. When $\mu_f > \mu_R$ the model ay then think the model to account assumption for snow avalanches. i friction angle from the front to ue, say, to the deposition of some valanches.

l resistance, μ_f and a linear decrease is fixed and not the value μ_R al velocity. Because the velocity of back we expect the model to have

fact that μ now does not depend e of mass. Only quantitative but no he models type I and II. In is of models 1 and 4 (or 5, 6) and ins from models 2 and 4 (or 5, 6).

Similarity solutions for granular avalanches

Table 1. Description of the basal friction coefficients for 3 model types. For type I models the friction coefficient μ depends on the spatial variable, for type II models μ is velocity dependent, whereas for type III models the friction coefficient is both position and velocity dependent

Models Type I	Models Type II	Models Type III
1) $\mu = \frac{\mu_f x}{x_f + \mu_f x} + \frac{\mu_R x}{x_R + \mu_R x}$ where μ_f, μ_R are constants independent of x and u .	3) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right)$ where μ_f, μ_R and u_0 are constants independent of x and u .	7) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where $f(u)$ is given by model 4), 5), or 6).
2) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where μ_f, μ_R and u_0 are constants independent of x .	4) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where $f(u) = 1 - \frac{u}{u_0}$.	8) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where $f(u)$ is given by model 4), 5) or 6).
3) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where μ_f, μ_R and u_0 are constants independent of x and u .	5) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where $f(u) = 1 - \frac{u}{u_0}$.	
4) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where μ_f, μ_R and u_0 are constants independent of x and u .	6) $\mu = \frac{\mu_f}{\mu_f + \mu_R} \left(1 - \frac{u}{u_0} \right) + \frac{\mu_R}{\mu_f + \mu_R} \left(1 - \frac{x}{x_0} \right)$ where $f(u) = 1 - \frac{u}{u_0}$.	

Fig. 3. The best-fit bounding-boxes of the image A002 in UW-I detected by the proposed algorithm in this paper. There are two parts on this image with different skew angles. The components in gray are the filtered-outs from the original image.

Fig. 3 shows the best-fit bounding-boxes of the sample image A002 in UW-I using the component-grouping algorithm in Ref. [13] and the proposed best-fit bounding-box algorithm in this paper. This image is scanned in the 2-up style, which contains an incomplete left page and a complete right page with sparse text. This sample image demonstrates the effectiveness of the proposed best-fit bounding-box algorithm in processing text blocks with distinct orientations, shapes and populations.

Fig. 4 shows the result on another sample image A03I in UW-I. This sample image shows how the component-grouping results affect the results of the best-fit bounding-box algorithm. On the incomplete right page, some of the noises in the dark spine area are wrongly grouped into text blocks by the used grouping function [13]. Therefore, some of the best-fit bounding-boxes are not the best for those text blocks. For this situation, the proposed best-fit bounding-box algorithm is not at fault.

It is also interesting to see how the proposed best-fit bounding-box algorithm works on the individual characters. Compared to the text blocks whose rectangular borders are well defined, the single characters are small and their contours are not intended to reflect the rectangular shape of their glyphs. This means that there may be more than one local minimum in the areas of the probing bounding-boxes, and the maximum to minimum ratio is not large enough to guarantee meaningful best fit. The image in Fig. 5 shows that in the normal style, the majority of the best-fit bounding-boxes are coincide with the page orientations, even in italic style.

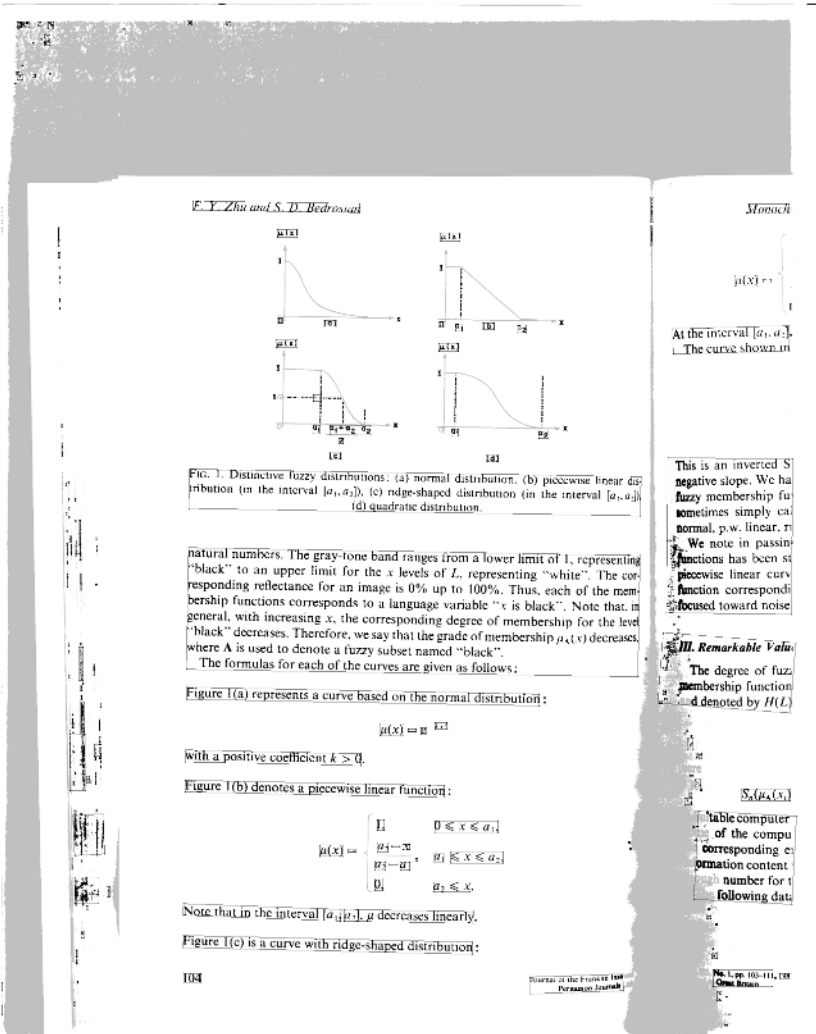


Fig. 4. The best-fit bounding-boxes of the image A03I in UW-I. This image shows that the grouping results have direct impact on the best-fit bounding-boxes. Some blocks in right half contain unfiltered noises that distorted the correct detection of the best-fit bounding-boxes.

Abstract

This paper describes a fast and effective component grouping algorithm for some layout analysis problems of natural document images. Based on an empirical function, it discovers neighborhood relations of the component level. Time-consuming algorithms that operate globally among components may obtain significant performance boost from the localization information that this grouping algorithm provides. Some of the applications using this grouping function are also demonstrated.

1. Introduction

Component-based methods for natural document images analyze regions information about the global structure or layout of the documents. There are several popular approaches for layout analysis. The bottom-up approaches start from the detected connected components and move up to higher-level structures of word, text line and paragraph [1][2]. The top-down approaches work along the opposite direction from the page level down to lower-level structures [3][4]. Applications may need to work on only some levels of structures.

Fig. 5. A clip of an article with the best-fit bounding-boxes of its individual characters in italic and normal styles

In many component-based algorithms for document image analysis, the components are abstracted to single points, called fiducial points by Spitz [7], which are sometimes chosen to be the bottom centers or other points along the borders of the upright bounding-boxes of the components [5]. The downside of these choices of the fiducial points is that they are not rotation-invariant. To solve this problem, the best-fit bounding-boxes of the components should be used so that the fiducial points at the borders become rotation-invariant. It is even more applicable for the Chinese and other East Asian languages whose font glyphs are visually rectangular.

3 An Application: Skew Estimation

One of the direct applications of the best-fit bounding-boxes is the multi-skew estimation of a page. A page usually contains several text blocks. The best-fit

bounding-boxes of the text blocks may provide important hints for estimating the orientations and locations of the individual blocks as well as the skew of the page as a whole.

There are many established skew estimation methods in literature [4]-[12], but this best-fit bounding-boxes based skew estimation method has the advantage of detecting and locating multiple skews in the same image. Furthermore, this method can be used to detect non-text blocks such as the rectangular graphical inserts or the tables with rectangular borders.

3.1 Weighted Skew of Page

A document image may have more than one part, such as when two facing pages are scanned in the same image. In such a case, each part has text blocks whose best-fit bounding-boxes differ slightly. The images in Fig. 3 and Fig. 4 raise the question of how the skew of a page can be estimated from the individual blocks in the same image.

We take a convolution-based approach [12] to locate the peak values from the resultant orientation histogram of the text blocks. Given a document image, the detected orientation of the best-fit bounding-boxes are accumulated in an accumulator array that has $N_{bin} = 9000$ bins. This represents an angle range of $[-90^\circ, 90^\circ]$, with an angle resolution of 0.02° per bin.

Given the best-fit bounding-box of a text block with skew angle θ and size n (number of components), the bin index of the text block in the accumulator array is $4500 + \theta / 0.02$, and the increment in this bin is the square-root of the size n . This choice on one hand gives larger weights to the larger text blocks in a hope that the shapes of these larger blocks are closer to rectangles and their true orientation can be more reliably approximated by their best-fit bounding-boxes. On the other hand, the influences of excessively long detected edges can be limited to some extent.

As shown in Fig. 6 (left), the skew histogram of the blocks in the sample image A03I in Fig. 4 has a cluster of discrete values around the largest peak. For the images that contain two pages, such as the sample image A002 in Fig. 3, there will be more than one cluster in the histogram. In order to distinguish the individual clusters that represent different multiple skews and at the same time weight the distribution within the clusters, the histogram is convolved by Eq. (4) with a finite, symmetric kernel generated from an un-normalized Gaussian distribution as in Eq. (3), where σ_{bin} is the variance and μ is a positive integer that represents the half-size of the kernel.

$$k[j] = \exp\left[-\frac{(j-\mu)^2}{\sigma_{bin}^2}\right] \quad (3)$$

$$h_{convol}[i] = \sum_{j=-\mu}^{\mu} h[(i-j-\mu+N_{bin}) \bmod N_{bin}] k[j+\mu] \quad (4)$$

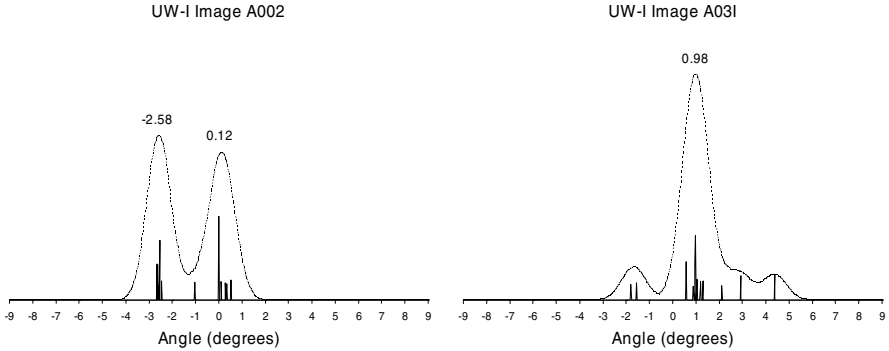


Fig. 6. The orientation histograms of the best-fit bounding-boxes of the images A002 (in Fig. 3) and A03I (in Fig. 4) in UW-I. An un-normalized Gaussian kernel ($\sigma = 0.5^\circ$) as in Eq. (3) is used to smooth the histograms.

In Eqs. (3) and (4), the half-size μ of the Gaussian kernel is set to $3\sigma_{bin}$. The value of σ_{bin} is set to 25 bins, which corresponds to 0.5° . The value of μ is 75 bins. The size of the kernel is 151 bins. The modulo operator in Eq. (4) indicates the wrapping of values at the two endpoints of the histogram.

The convolved histogram in Eq. (4) is used to search for the prominent peaks that correspond to the dominant skew angles in an image. For the raw histogram in Fig. 6 (left), the convolved histogram is shown in Fig. 6 (right). This convolved histogram also shows the desired property of the chosen kernel in Eq. (3), which combines the smoothing and area subtraction in one.

3.2 Suite Test Using UW-I

To evaluate the effectiveness and robustness of the proposed best-fit bounding-box based skew estimation method, the real document images from the University of Washington English Document Image Database I (UW-I) are used. In this database, total 979 images are scanned from real printed journals. Many images contain large area of disjoint, non-textual components that are the results of binarization on photographic objects, or the artifacts of the scanning process. Using a widely used document database makes it possible for different groups of researchers to evaluate and compare their algorithms with a common benchmark.

Fig. 7 (top) gives the regression analysis of this suite test, where the linear correlation coefficient is 88.4%. Fig. 7 (bottom) shows the accumulated percentage of samples versus the absolute detection error. It can be seen from these test results that this best-fit bounding-box based skew estimation method performs very well for real world document images.

The execution speed of the proposed skew estimation method is very fast based on the experimental results. It takes less than 2 seconds in average to process a sample image on the Java 5 platform in a 3GHz Pentium 4 computer.

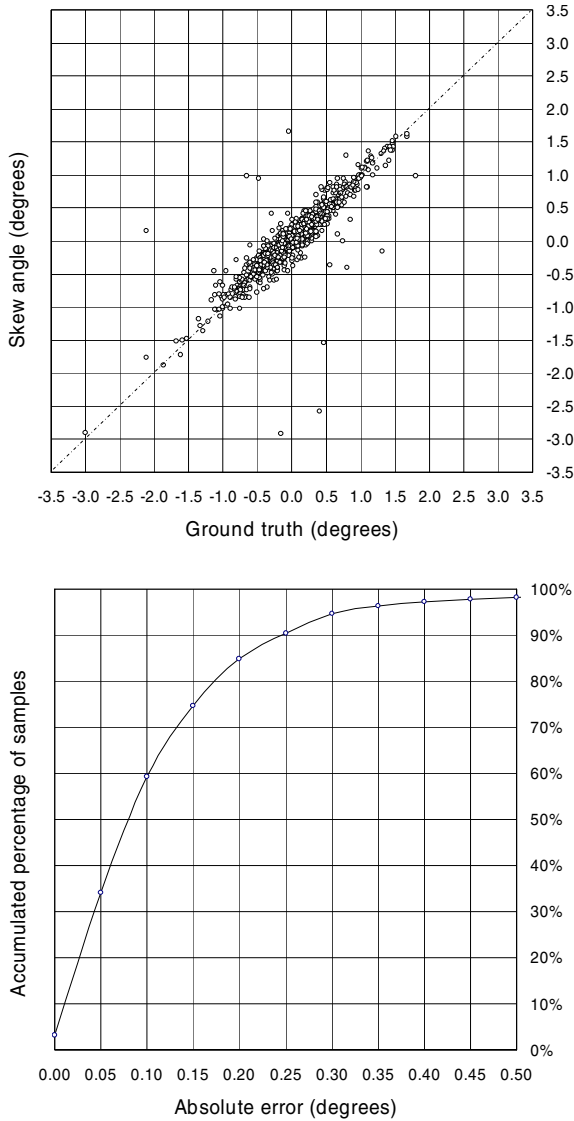


Fig. 7. The regression analysis (top) and the accumulated percentage of samples (bottom) of the 979 real document images in UW-I

4 Conclusions

This paper presents an algorithm for finding the best-fit bounding-boxes of text blocks in scanned document images. This algorithm is based on the principle that the best-fit bounding-box of text block has the minimum area compared to the bounding-boxes with other orientations. This algorithm is reliable, efficient and applicable in a

number of applications. One of the applications is the visual marking of the text segmentation results so that the less accurate and less eye-pleasing but simple and widely used upright bounding-boxes can be substituted by the best-fit bounding-boxes produced by the proposed algorithm with very low computational cost. Another potential application is the consolidation of the component grouping results, which merges the small groups with larger ones if they are fully contained in the best-fit bounding-boxes of the larger ones. This is one of the remedies for under-grouping, which is often encountered in text segmentation using bottom-up approaches. The application in page skew estimation has been singled out for detailed investigation, as it is an interesting application as well as an indirect way of verifying the best-fit bounding-box algorithm. The suite test using the UW-I database shows that the skew estimation method based on the proposed best-fit bounding-box algorithm is accurate, robust and efficient enough for practical use.

References

1. G. Nagy, "Twenty Years of Document Image Analysis in PAMI", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22 (1), pp. 38-62, January 2000.
2. L. O'Gorman, and R. Kasturi. Document Image Analysis. IEEE Computer Society Press, Los Alamitos CA, 1995.
3. R. Cattoni, T. Coianiz, S. Messelodi, and C. M. Modena, "Geometric Layout Analysis Techniques for Document Image Understanding: a Review", ITC-IRST Technical Report #9703-09, 1998
4. W. Postl, "Detection of Linear Oblique Structures and Skew Scan in Digitized Documents", Proceedings of the 8th International Conference on Pattern Recognition, pp. 687-689, Paris, October 1986.
5. H. S. Baird, "The Skew Angle of Printed Documents", Proceedings of the SPSE 40th Annual Conference and Symposium on Hybrid Imaging Systems, pp. 21-24, Rochester, NY, May 1987.
6. Y. Nakano, Y. Shima, H. Fujisawa, J. Higashino and M. Fujinawa, "An algorithm for skew normalization of document images," Proceedings of the 10th International Conference on Pattern Recognition, pp. 8-13, Atlantic City, New Jersey, 1990.
7. A. L. Spitz, "Skew Determination in CCITT Group 4 Compressed Images", Proceedings of the 1st Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, pp. 11-25, 16-18 March 1992.
8. S. N. Srihari, and V. Govindaraju, "Analysis of Textual Images Using the Hough Transform", Machine Vision and Applications, Vol. 2 (3), pp. 141-153, 1989.
9. S. Hinds, J. Fisher, and D. D'Amato, "A document skew detection method using run-length encoding and the Hough transform", Proceedings of the 10th International Conference on Pattern Recognition, pp. 464-468, Atlantic City NJ, 17-21 June 1990.
10. S. Chen, and R. M. Haralick, "An Automatic Algorithm for Text Skew Estimation in Document Images Using Recursive Morphological Transforms", Proceedings of IEEE International Conference on Image Processing, pp. 139-143, Austin TX, 13-16 November 1994.
11. H. K. Aghajan, and T. Kailath, "SLIDE: Subspace-Based Line Detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16 (11), pp. 1057-1073, November 1994.

12. B. Yuan, and C. L. Tan, "Fiducial line based skew estimation", *Pattern Recognition*, Vol. 38 (12), pp. 2333 – 2350, December 2005.
13. B. Yuan, and C. L. Tan, "A Multi-Level Component Grouping Algorithm and Its Applications", *Proceedings of the 8th International Conference on Document Analysis and Recognition*, pp. 1178-1181, Seoul Korea, 29 August - 1 September 2005.
14. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, "Numerical Recipes in C : The Art of Scientific Computing", Second Edition, Cambridge University Press, 1992.