

A Part-Based Skew Estimation Method

Soma Shiraishi, Yaokai Feng, and Seiichi Uchida

Kyushu University, Fukuoka, Japan

shiraishi@human.ait.kyushu-u.ac.jp, {fengyk, uchida}@ait.kyushu-u.ac.jp

Abstract—In this paper we propose a part-based skew estimation method which is more robust to larger varieties of text images, such as camera-captured scene images. Specifically, the skew angle at each local part of the input image is estimated independently by referring the local part of upright character images stored as a database. Then the global skew angle is estimated by aggregating the estimated local skews. The proposed method does not assume that characters are laid-out in straight lines and thus have more robustness to the varieties of text images than conventional methods. The experimental results show the advantage of the proposed method over the conventional methods under several conditions.

I. INTRODUCTION

Skew estimation is one of the most important preprocessing steps for OCR. Recently, it is required to be more robust to deal with various target images, such as freely laid-out text images and camera-captured scene text images (e.g., presentation slides projected on a screen), in addition to regular business document images by scanner.

There are several conventional skew estimation methods [1], such as the projection profile method, the Hough transform method, and the nearest-neighbor method. All of three methods assume that the characters in images are laid out in straight lines, therefore they estimate the text skew by finding text lines by their own approaches and then measuring their angles. Accordingly, those three methods may fail to estimate the correct skew angle in the case where text lines are randomly or irregularly laid-out. One such example is document images which include many mathematical equations. Another example is camera-captured scene images where some texts are included in a scattered manner.

In this paper, we propose a novel skew estimation method, where local parts extracted from each character are the unit of estimating local skew angles. Since the proposed method relies on the local skew estimation, it can deal with irregularly laid-out documents. Furthermore, since local parts can be detected from the input image without binarization, we can expect more robustness to occlusions and complex backgrounds than using the full shapes of characters. As the extraction and the description of local parts, we will use Speeded-Up Robust Features (SURF) detector [2], while we can use another local feature extraction method.

The rest of the paper is organized as follows. In Section II, the conventional methods are first reviewed briefly. In Sec-

tion III, the principle of the proposed method is described. In Section IV, experimental results by the proposed method and the conventional methods are shown. Section V draws the conclusion of the paper.

II. CONVENTIONAL METHODS

There are several conventional methods for skew estimation. In the followings, the three most representative methods, that is, the projection profile method, the Hough transform method, and the nearest-neighbor method, are reviewed briefly. A conventional part-based method is also reviewed.

A. The Projection Profile Method

The projection profile method (e.g., [3]) utilizes a histogram acquired by accumulating the number of black pixel (in a binarized image) along parallel sample lines through the document. For example, for the horizontal writing, the projection profile taken horizontally along rows have the narrowest peaks. In the most straightforward method, projection profile is calculated for each expected orientation, and one with the keen peaks shows the skew angle. A modified version of projection profile proposed by Akiyama, et al. [4] first separates the document into several “swaths”. Projection profile is then calculated for each of the swaths. By detecting the shift which maximizes correlation between adjacent projection profile, the skew angle is estimated.

B. The Hough Transform Method

Hough transform can be used to estimate a skew angle (e.g., [5]). If the characters are laid out in straight lines, the centers of gravity of the characters align in straight lines accordingly. By simply finding the slope of lines which go through those the centers of gravity using Hough transform, the text skew can be estimated [6]. Amin and Fischer[7] first find connected components in an input image and group them together according to the distance between them. Each of the grouped areas is now divided into several swaths whose widths are about the same size as a connected component. Subsequently a centroid of the connected component at the very bottom in each swath is selected, and in each of the area, those selected centroids are used for the Hough Transform.

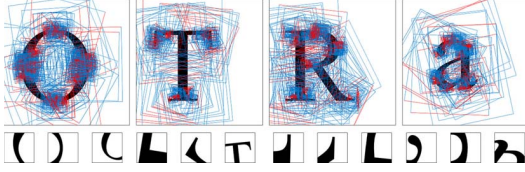


Figure 1. Local parts detected by SURF. A red line in each square was used to indicate the orientation.

C. The Nearest-Neighbor Method

In the nearest-neighbor method, connected components are determined first. Then, for each connected component, the nearest neighbors components are found. In [8], the histogram of the angles to the nearest neighbor components is created. The final skew is estimated as the peak of the histogram. O’Gorman [9] has used not only 1 but also k -nearest neighbors (where k is usually 5) to first make a rough skew estimation. After getting rid of the between-lines nearest neighbors, more accurate estimation is calculated using only within-line nearest neighbors.

D. The Conventional Part-Based Method

One solution to the problem of the above conventional methods has been proposed in [10]. Similarly to the proposed method, this is also a part-based skew estimation. Specifically, this method is based on skew estimation of each connected component (\sim a character) by comparing it to the stored shape of reference connected component images. Finally, the skew angle of “each character” can be found and most frequent local skew angle is chosen as the global skew. The drawback of [10] is that it totally relies on the connected components being extracted. Since the process of extracting connected components depends on the quality of binarization, the skew estimation will fail if binarization fails to extract connected components of characters accurately.

III. PART-BASED SKEW ESTIMATION

A. The Key idea

The key idea of the proposed method is to estimate the local skew angle for each of local parts. Here, “local part” can be just a part of a character. For example, the bottom end of “T” can be a local part. Roughly speaking, we can estimate the skew of this local part by observing it. If this “T” is printed in a Times-Roman font, we can roughly estimate the skew by observing the tilt of the serif. Even if it is in a Gothic font, there is still a high possibility to have a correct estimation.

The local part is better to be around corner or ending points of a character. This is because a local part just containing a straight line segment will be difficult to estimate its skew. Accordingly, recent so-called “keypoint detectors”, such as SURF and SIFT, are suitable for this purpose. Figure 1 shows an example of local parts selected by

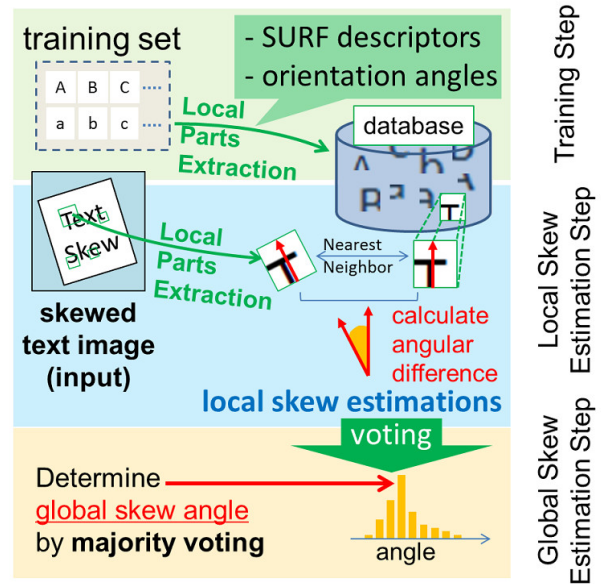


Figure 2. Overview of the proposed method.

SURF [2]. Many local parts are detected around the corner and large curvature areas. In contrast, straight line areas are less detected as local parts.

Note that even if we have several local parts whose skew is difficult to be estimated, their effects are not serious. In fact, we can have many local parts in a target image as shown in Fig. 1, and by aggregating all the local skew estimations, the erroneous estimations can be canceled.

A benefit of using local parts is that no preprocessing is necessary for preparing them. For example, any binarization process, which is required for connected component analysis, is not necessary. In fact, the above keypoint detectors are directly applicable to grayscale images for detecting less ambiguous local parts, such as corners.

As shown in Fig. 2, the proposed method is comprised of three steps; that is, training step, local skew estimation step, and global skew estimation step. Their roles are, to prepare a sufficient number of reference local parts, to estimate the local skew by comparing the target and reference local parts, and to aggregate the estimate local skews to have a reliable global skew angle of the entire image.

B. Training Step

In the training step, a database of reference local parts is prepared. First, upright character images (i.e., font images) are prepared as training dataset. If necessary, multiple font images are prepared. Then local parts on each character image are detected automatically with some keypoint detector. In this paper, we will employ SURF [2]. Again, Fig. 1 is a detection example by SURF. Each local part is described by as a 128-dimensional feature vector and its elements are gradient information within the local part.

There are three important properties of SURF and they are essential to estimate the local skew in the next step.

- First, the detected local part positions are the same regardless of the scale and the skew of the target image.
- Second, it can determine the “dominant orientation” at each local part based on image gradient. If the target image undergoes a skew (i.e., a rotation) of θ , the dominant orientation of a local part is also changed by θ from the original orientation.
- Third, SURF feature vector is skew (and scale) invariant. SURF adaptively changes the orientation and scale of its local part and describe the part as a vector. Consequently, the resulting vector is constant theoretically regardless of the skew and scale of characters in the target image.

As shown in Fig. 2, all of the SURF feature vectors of training images and their dominant orientations are paired and stored in a database. Each paired entry is a considered as an instances and used for the local skew estimation.

C. Local Skew Estimation Step

The skew angle of each local part of an input image is estimated by referring the database. Specifically, as shown in Fig. 2, the nearest neighbor (measured by Euclidean distance in the feature vector space) for the input local part is first found from the database. Because of the invariance of the SURF feature vector, we can expect that the input local part and its nearest neighbor are the same local part of a certain character. Then, recalling the second property of SURF explained in III-B, the skew angle of the input local part is estimated just by checking the difference of the dominant orientations of the input local part and its nearest neighbor.

D. Global Skew Estimation Step

The global skew angle is finally estimated by aggregating the estimated local skew angles. An important point on the aggregation is that some estimated local skew angles are largely deviated from the true global skew angle. This is because, for example, the nearest neighbor in the database is sometimes a different local part due to the ambiguity of local parts.

As a robust aggregation method which does not affected by the large deviations, we use a simple majority voting scheme as shown in Fig. 2. Specifically, each local skew angle is voted into its corresponding bin of an angle histogram. The width of each bin is predetermined according to the skew sensitivity of the succeeding character recognition. Here, the bin width is 1 degree and thus the histogram has 360 bins in total. The global skew angle is estimated as the angle of the bin with the maximum votes.

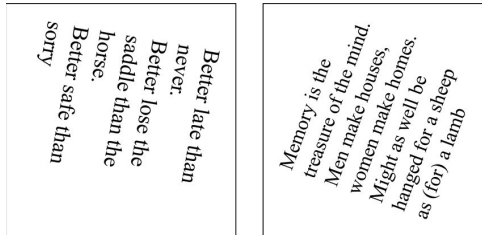


Figure 3. Examples of the test set used in basic performance test.

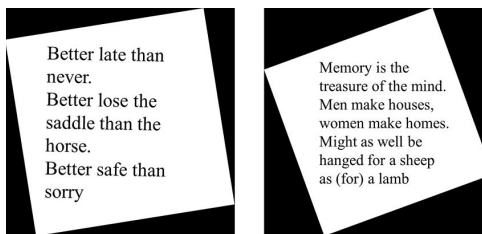


Figure 4. Deskewed text images of Fig. 3 by the proposed method.

Table I
SKEW ESTIMATION ACCURACY ON EQUATION IMAGES BY VARIOUS METHODS.

| Method | Proposed | PP | HT | NN |
|------------------|----------|-----|-----|-----|
| Average Error[°] | 1.2 | 2.6 | 1.7 | 1.9 |
| Maximum Error[°] | 4 | 14 | 16 | 8 |

IV. EXPERIMENTS AND RESULTS

A. Basic Performance Test

An experiment has been conducted to observe the basic performance of the proposed method on 200 text-only images¹. As training set we chose characters 0-9, lowercased a-z, and capitalized A-Z in Times New Roman. The size of characters is 1600×1600 squared pixels. Examples of the test images are shown in Fig. 3. About 570 SURF keypoints are extracted from each image on an average. The test set consists of 50 images each skewed to -170 degrees, -100 degrees, 0 degree, 5 degrees, and 100 degrees. The size of characters in those images were around 52×52 squared pixels.

The proposed method had achieved an average error of 0.3 degree. Even in the worst case, its error was within ± 2 degrees. Examples of the deskewed result is shown in Fig. 4. The errors are due to the inaccuracy of SURF to detect exactly the same parts from upright character images and skewed text images, and also due to the inconsistency of calculated orientation caused by adjoining characters in the input image. There is a possibility that we can improve the accuracy of the proposed method by choosing more suitable features.

¹All the test images used in this paper are available at <http://human.ait.kyushu-u.ac.jp/~shiraishi/testimages-DAS2012.zip>

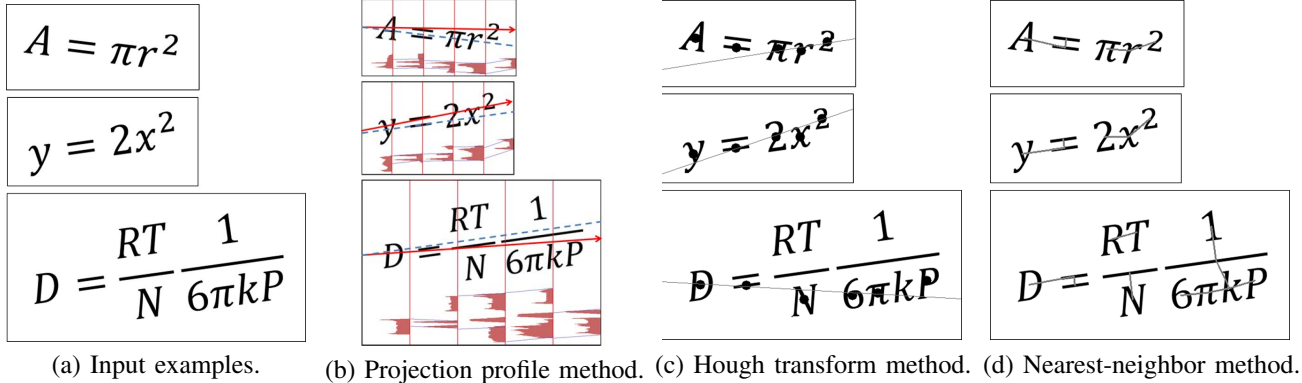


Figure 5. Skew estimation results of the conventional methods.

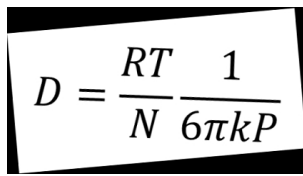


Figure 6. Deskewed image by the proposed method.

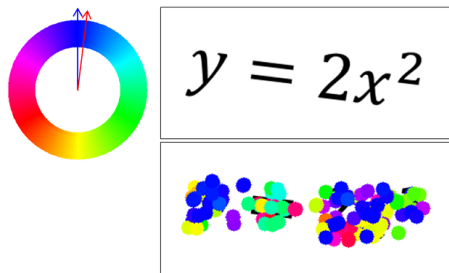


Figure 7. Local skew estimations are shown by corresponding colors.

B. Comparison with Conventional Methods

We implemented three conventional methods shown in Section 2 ([4], [7], [8]). A weakpoint with those conventional methods is that since all of them assume the characters in images are laid out to form straight lines, they sometimes fail to estimate the text skew angle in images where characters are rather randomly laid out.

As a simple example of such text images, mathematical equations was employed in the second experiment. In this experiment, 60 test images of different mathematical equations were used for comparing the performance of the proposed method and the other three methods. Examples of the test set are shown in Fig. 5(a).

Table I shows the accuracies; the proposed method had more accuracy than all of the three conventional methods on the test set. An actual estimation process of the conventional methods are illustrated in 5 respectively. In Fig. 5(b) lines between histograms show the shift that makes the correlation largest. In Fig. 5(c) selected text lines are shown in each of the images. It is seen that text lines are not detected correctly

Table II
RESULT OF SCENE TEXT

| Image | (a) | (b) | (c) | (d) | (e) | (f) |
|---------------------|-----|------|------|------|-----|------|
| Skew Angle[°] | -12 | 46 | 24 | -26 | -60 | -7 |
| Estimation Error[°] | 1 | 0 | 1 | 1 | 2 | 0 |
| # of detected parts | 860 | 1477 | 2060 | 1878 | 294 | 5459 |

due to the characters off the actual text lines. Fig. 5(d) shows the lines connecting nearest components. Among the three examples, the bottom example is the most difficult for those three methods. In contrast, as shown in Fig. 6, the proposed method have the best deskew result.

Figure 7 shows the distribution of the local skew estimation results by their corresponding colors. The red arrow in the color circle indicates the skew angle of the text. It is no surprise and yet interesting to see that, on ‘y’ and ‘2x²’, the blue dots that correspond to the correct skew angle dominates. Note that on ‘=’, the blue color do not appear since the database do not contain ‘=’ in this experiment. This also indicates the robustness of the proposed method against the lack of reference local parts.

C. Scene Images

In this experiment, the test set are 5 scene images captured with a digital camera and 1 synthetic poster image. Each of the scene images are 3456×2592 squared pixels, and the poster image is 723×720 . The sizes of characters are not fixed. The skew angle is first correctly measured by hand to evaluate the output of the proposed method. The table II shows the result of the experiment. On the test set the accuracy of the proposed method was ± 2 degrees, and average of error was 0.43 degree. In the Fig. 8(a), (c), (d), and (f), in spite of the occlusions on text regions, the proposed method can estimate the skew angle correctly. In (b), and (e) the method estimated the skew correctly even with complex backgrounds in images.

Figure 9 shows the distribution of locally estimated angles by colors. It is very interesting to see that, in the character region, the color corresponding to the correct skew angle

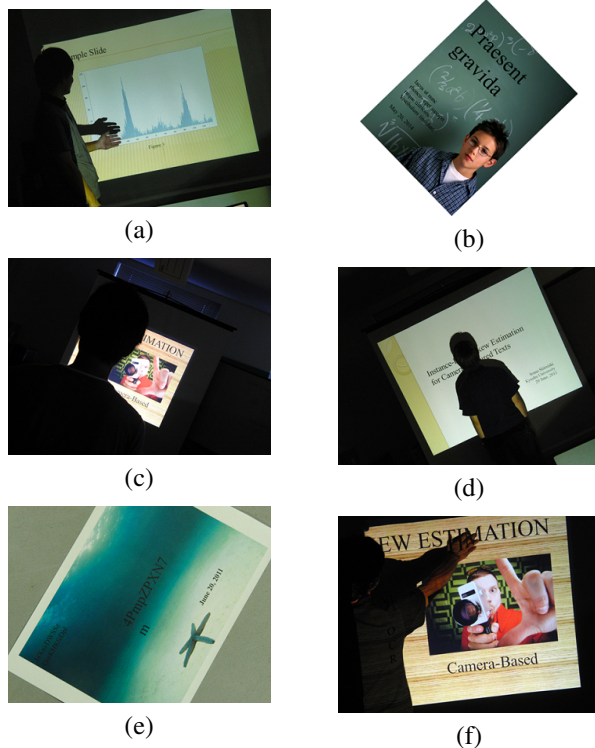


Figure 8. Test set in experiment C.

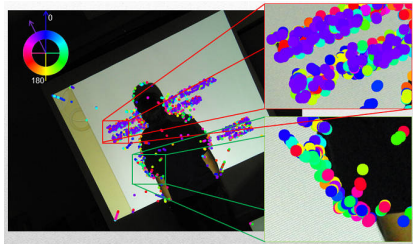


Figure 9. Local skew estimations are shown by colors. Correspondence relation between angles and colors are shown on the top left.

dominates the area. On the other hand, in the non-character region, variety of colors are randomly seen. This fact means that, with majority voting, the correct estimation are obtainable even in images with non-character background.

Figure 10 shows the comparison result between the proposed method and the three other methods. Test set was 76 presentation slide images (different from the ones in Experiment B) that are similar to the ones in Fig. 8 (without occlusions). The result indicates the constant accuracy of the proposed method against other methods. This stability is because of the fact that the accuracy of the proposed method does not rely on the quality of binarization and detection of character regions.

V. CONCLUSION

In this paper we had proposed a part-based skew estimation method. An important property of the proposed method is that it does not assume that text skew can be estimated by

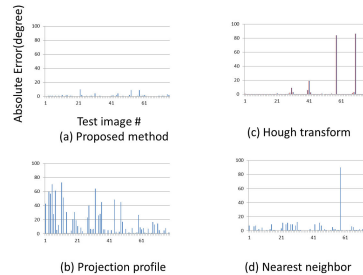


Figure 10. Comparison with conventional methods.

the angle of text lines. Instead, the method utilizes the local parts of characters as a fundamental unit of skew estimation. Consequently, it is effective on document images without explicit (and long) text lines. The experimental results have shown the advantage of the proposed method over the conventional methods. The results have also shown that the proposed method is applicable to scene images with varieties of occlusions and complex backgrounds.

In the future, the accuracy of the method can be improved by choosing more suitable feature detector and descriptor. Another improvement can be made by refining the database since some of the local parts have less accuracy in calculated orientation than others. Extension of nonuniform skew, especially, perspective distortion is also an important future work.

REFERENCES

- [1] J. Hull, "Document Image Skew Detection: Survey and Annotated Bibliography," *Document Analysis Systems II*, World Scientific, pp. 40–64, 1998.
- [2] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," *Proc. ECCV*, 2006.
- [3] W. Postl, "Detection of Linear Oblique Structures and Skew Scan in Digitized Documents," *Proc. ICPR*, pp.687-689, 1986.
- [4] T. Akiyama and N. Hagita, "Automated Entry System for Printed Documents", *Pat. Recognit.*, 23(11), 1990.
- [5] S. N. Srihari and V. Govindaraju, "Analysis of Textual Images Using the Hough Transform," *Mach. Vis. Appl.*, 2, pp.141-153., 1983.
- [6] L. O’Gorman and R. Kasturi, *Document Image Analysis*, IEEE CS Press, 1997.
- [7] A. Amin and S. Fischer, "A Document Skew Detection Method Using the Hough Transform", *Pat. Anal. Appl.*, vol. 3, no. 3, pp.243-253, 2000.
- [8] A. Hashizume, P-S. Yeh, and A. Rosenfeld, "A Method of Detecting the Orientation of Aligned Components," *Pat. Recognit. Lett.*, 4, pp. 125-132, 1986.
- [9] L. O’Gorman, "The Document Spectrum for Page Layout Analysis," *TPAMI*, 15(11), pp. 1162-1173, 1993.
- [10] S. Uchida, et al., "Skew Estimation by Instances," *Proc. DAS*, pp. 201–208, 2008.