

Part-Based Skew Estimation for Mathematical Expressions

Soma Shiraishi, Yaokai Feng, and Seiichi Uchida

shiraishi@human.ait.kyushu-u.ac.jp

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

Abstract

We propose a novel method for the skew estimation on text images containing mathematical expressions which can be applied to various characters layouts. Current OCR systems are not capable of recognizing skewed characters in images correctly, and hence skew correction in such images is essential for character recognition. Conventionally methods such as projection profile methods, Hough transform methods, and nearest neighbor methods are used for skew estimation. They assume characters form straight text lines in images. By rotating the images so that those lines become horizontal, the skew of characters is estimated. Mathematical expressions, however, are difficult to apply the conventional methods to because equations often mislead them by containing suffixes, fractions, and matrices to form false text lines. The proposed method, on the other hand, is a part-based method that utilizes the local features of characters. Specifically, the system first creates a database by extracting the local features from upright character images and storing them. Subsequently it also extracts the local features from an input text image. The skew angle at each local part of the input image is estimated independently by referring to the database. Then the global skew angle is estimated by the majority voting on the estimated local skews. This skew estimation process enables the proposed method to be free from the assumption of the straight text lines in images. Therefore the proposed method can be applied to text images with mathematical expressions that generally contain various layouts of characters and symbols. Through experiments, we tested the performance of this method on the images that contain mathematical equations.

1 Introduction

When digitizing documents that contain mathematical expressions, text skew is one of the problems. There are several conventional skew estimation methods [1], such as projection profile methods, Hough transform methods, and nearest-neighbor methods. Essentially all of these methods correct the skew in the same manner, that is, they estimate the angle of the text lines, and correct the angle for them to be horizontal. These methods have high accuracy in many cases, but, mathematical expressions sometimes contain suffixes, fractions, matrices, and so on, which creates a false text lines in documents, and make it difficult for the methods to estimate the correct skew angle.

In this paper, we apply a part-based method that has been proposed in our recent work[2]. This method is novel for being a part-based skew estimation method on mathematical expressions that utilizes local parts of characters unlike the conventional methods explained in the following section. 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. For the extraction and the description of local parts, we will use the Speeded-Up Robust Features (SURF) detector [3]. The skew addressed by this paper is the rotation of characters.

The rest of the paper is organized as follows. In Section 2, the conventional methods are first reviewed briefly. In Section 3, the principle of the proposed method is described. In Section 4, experimental results are shown. Section 5 draws the conclusion of the paper.

2 Conventional methods

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

2.1 The Projection Profile Method

The projection profile method (e.g., [4]) uses a histogram acquired by accumulating the number of ON pixel in a binarized image along parallel sample lines through the image. The narrowest peaks can be acquired when the projection profiles are taken horizontally along rows on horizontal writings. In the most straightforward method, projection profile is calculated for each expected orientation, and one that gives the keenest peaks shows the skew angle. A modified version of projection profile has been proposed by Akiyama, et al. [5]. The method first separates the document into several “swaths”. Projection profile is then calculated for each of the swaths. By finding the shift which maximizes correlation between adjacent projection profile, the skew angle can be estimated.

2.2 The Hough Transform Method

It has been shown that Hough transform can be used to estimate a skew angle (e.g., [6]). If the characters form straight text lines, the centers of gravity of the characters align nearly straight accordingly. By simply finding the slope of the lines made of the centers of gravity using Hough transform, the text skew can be estimated. Amin and Fischer[7] first find connected components in an input image and make groups of mutually close connected components. The distance threshold for the grouping is a parameter. 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.

2.3 The Nearest Neighbor Method

The nearest neighbor method can be explained with four steps, determining connected components, finding the nearest neighbors for each of the components, calculating the angles between the components of the nearest neighbor pairs, and finally making a histogram of the angles to find the most frequent value. O’Gorman [9] has proposed a modified version where not only 1 but also k -nearest neighbors (where k is usually 5) are found to first make a rough skew estimation. After eliminating the between-lines nearest neighbors, more accurate estimation is calculated using only within-line nearest neighbors.

2.4 The Conventional Part-Based Method

As a solution to the problem of the above conventional methods, another method has been proposed in [10]. This is a part-based skew estimation similarly to the proposed method. Specifically, this method is based on skew estimation of each connected component (\sim a character) by comparing it to preliminarily stored character shapes. 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 extraction quality of connected components. The process of extracting connected components depends on the quality of binarization, and therefore the skew estimation fails if binarization fails to extract connected components of characters accurately.

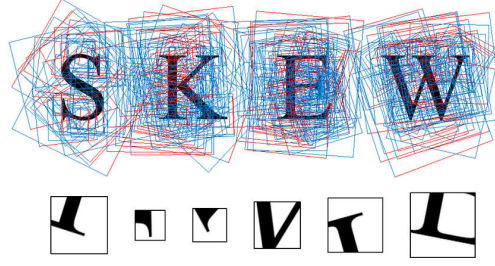


Figure 1: Examples of the local parts extracted by SURF

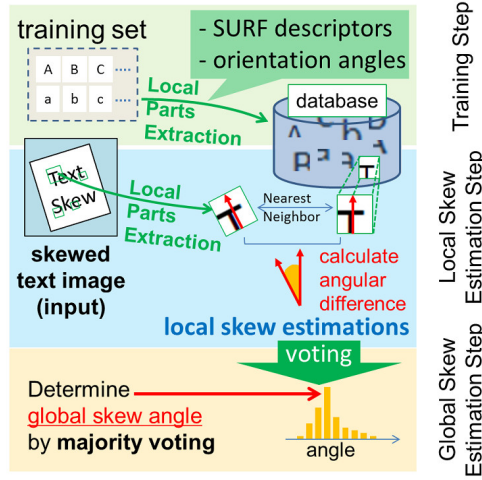


Figure 2: The principle of the proposed method.

3 The Proposed Part-based Skew Estimation

3.1 The Key Idea

The key idea is to estimate the local skew angle for each of local parts of characters. “Local part” can be defined in this paper as a part of a character such as the bottom end of “T”. Roughly speaking, we can estimate the skew of this local part by comparing it to a database. If “T” is printed in a Times-Roman font, we can roughly estimate the skew by observing the tilt of the serif part. This can possibly be stated to other fonts as well.

It is better that the local parts are around corners or ending points of a character. This is because a local part such as a simple straight line segment will be difficult to estimate its skew for it not having a distinctive shape. 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 SURF [3]. 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 estimate for the above reason, 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.

One of the benefits of using local feature detectors is that no preprocessing is necessary. 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, a training step, a local skew estimation step, and a global skew estimation step. Their roles are, respectively, 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.

3.2 Training Step

In the training step, reference local parts are stored into a database. First, upright character images (i.e., font images) are prepared as a training dataset. If necessary, multiple font images are prepared. Then local parts on each character image are detected automatically with a keypoint detector. In this paper, we will employ SURF [3]. Again, Fig. 1 is a detection example by SURF. Each local part is described as a 128-dimensional feature vector and its elements are gradient information within the local part.

There are three important properties of SURF which 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 vectors are skew (and scale) invariant. SURF adaptively changes the orientation and scale of its local part and describes 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 considered as an instance and used for the local skew estimation.

3.3 Local Skew Estimation Step

On an input image, local parts are extracted with a detector as well as the reference parts in the training step. The skew angle of each local part of the input image is estimated by referring to 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. (Of course when an image contains components, such as graphs and pictures, that are not in the database, this is not the case, but this will be canceled in the next step.) Then, recalling the second property of SURF explained in 3.2, 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.

3.4 Global Skew Estimation Step

The global skew angle is finally estimated by aggregating the estimated local skew angles from the previous step. An important result of the aggregation is that some estimated local skew angles deviate

$$A = \begin{bmatrix} a_{11} & \cdots & a_{14} \\ \vdots & \ddots & \vdots \\ a_{41} & \cdots & a_{44} \end{bmatrix} = \begin{bmatrix} 3.0 & 10.0 & 17.0 & 24.0 \\ 4.0 & 11.0 & 18.0 & 25.0 \\ 5.0 & 12.0 & 19.0 & 26.0 \\ 6.0 & 13.0 & 20.0 & 27.0 \end{bmatrix}$$

the velocity-stress time domain equations

$$\rho \frac{\partial V_x}{\partial t} = \frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xz}}{\partial z} + f_x$$

$$\rho \frac{\partial V_z}{\partial t} = \frac{\partial \sigma_{zx}}{\partial x} + \frac{\partial \sigma_{zz}}{\partial z} + f_z \quad (1)$$

where V_x and V_z are velocities in the x and z directions, respectively; σ_{xx} and σ_{yy} are normal stress tensors, and σ_{xy} is a shear stress tensor; f_x and f_y are external body forces in the x and z directions, respectively; and ρ is the bulk density.

the strain-velocity time domain equations

$$\frac{\partial \epsilon_{xx}}{\partial t} = \frac{\partial V_x}{\partial x}$$

$$\frac{\partial \epsilon_{zz}}{\partial t} = \frac{\partial V_z}{\partial z}$$

$$\frac{\partial \epsilon_{xz}}{\partial t} = \frac{1}{2} \left(\frac{\partial V_x}{\partial z} + \frac{\partial V_z}{\partial x} \right) \quad (2)$$

where ϵ_{xx} and ϵ_{yy} are normal strain tensors and ϵ_{xy} is the shear strain tensor.

the stress strain tensor relation in the frequency domain can be written

$$\sigma(\omega) = M(\omega)\epsilon(\omega) \quad (3a)$$

where M is the complex modulus matrix and ω is the angular frequency. The underlined terms correspond to tensors.

Figure 3: Test images for testing the basic performance.

considerably 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.

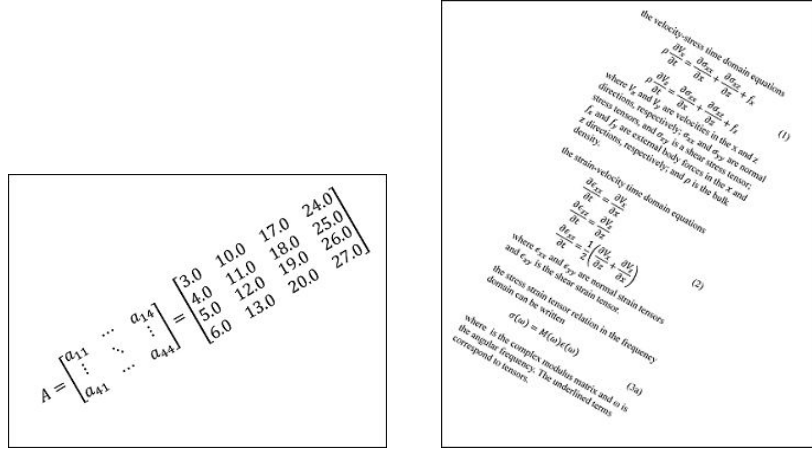
In order to suppress the effect of the local parts whose skew angles have large deviations, we use a simple majority voting scheme as shown in Fig. 2. Specifically, each local skew angle is voted into its corresponding bin of an angle histogram. The width of each bin is predetermined according to the skew sensitivity of the succeeding character recognition and also to the resolution of feature detector. 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.

4 Experiment

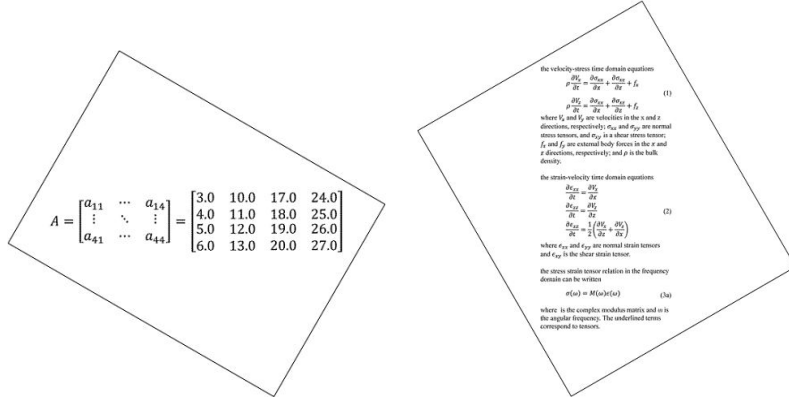
4.1 Basic Performance

In order to examine the performance of the proposed method on text images that contain mathematical expressions, an experiment was conducted. The test images were created by rotating each of two images shown in Fig. 3 to from 10 degrees to 350 degrees increasing the skew angles by 10 degrees. We had 72 test images in all. The size of the test images shown in Fig. 3 are 2768×692 (left), and 1984×3856 (right) respectively. Both of the images contain two kinds of font that are Times New Roman and Cambria Math, so we used upright alphabetical and numeric single character images of Both fonts as the training set.

The method had the mean absolute error of 1.1 degrees, and the errors were ± 3 degrees at a maximum. Several result images were shown in Fig. 4 The experiment has shown that the skew of characters in mathematical expressions can be corrected by the proposed method. There is no limitation on the skew angle where the proposed method is applicable.



(a)Test images.



(b)Result Images.

Figure 4: Skew correction results.

4.2 Comparison to the Conventional Methods

An experiment has been conducted in order to compare the performance of the proposed method to the conventional methods explained in Section 2, which are, a projection profile method, Hough transform method, and nearest neighbor method [5, 7, 8].

Five test images were generated on the computer (Fig. 5). Each images are rotated to from -15 degrees to 15 degrees increasing the rotation angle by one degree. We chose this range because the projection profile method and the Hough transform method we implemented for this experiment was said to cover that range [11].

Figure 6 shows the result of the experiment. We found that the proposed method had a rather constant accuracy with the error of ± 6 degrees at the largest on the test images. On the other hand the conventional methods' performances depended on the test images, i.e., the conventional methods worked well on some test images while they worked poorly on others. The conventional methods did not work well when connected components were mistakenly regarded to be in a text line with wrong connected components.

An example of failed skew estimations made by the Hough transform method is shown in Fig. 7. The Connected components were grouped rather vertically than horizontally due to the unique layout of the matrices, and the slopes of the text lines are incorrect. This error can, of course, be dealt with

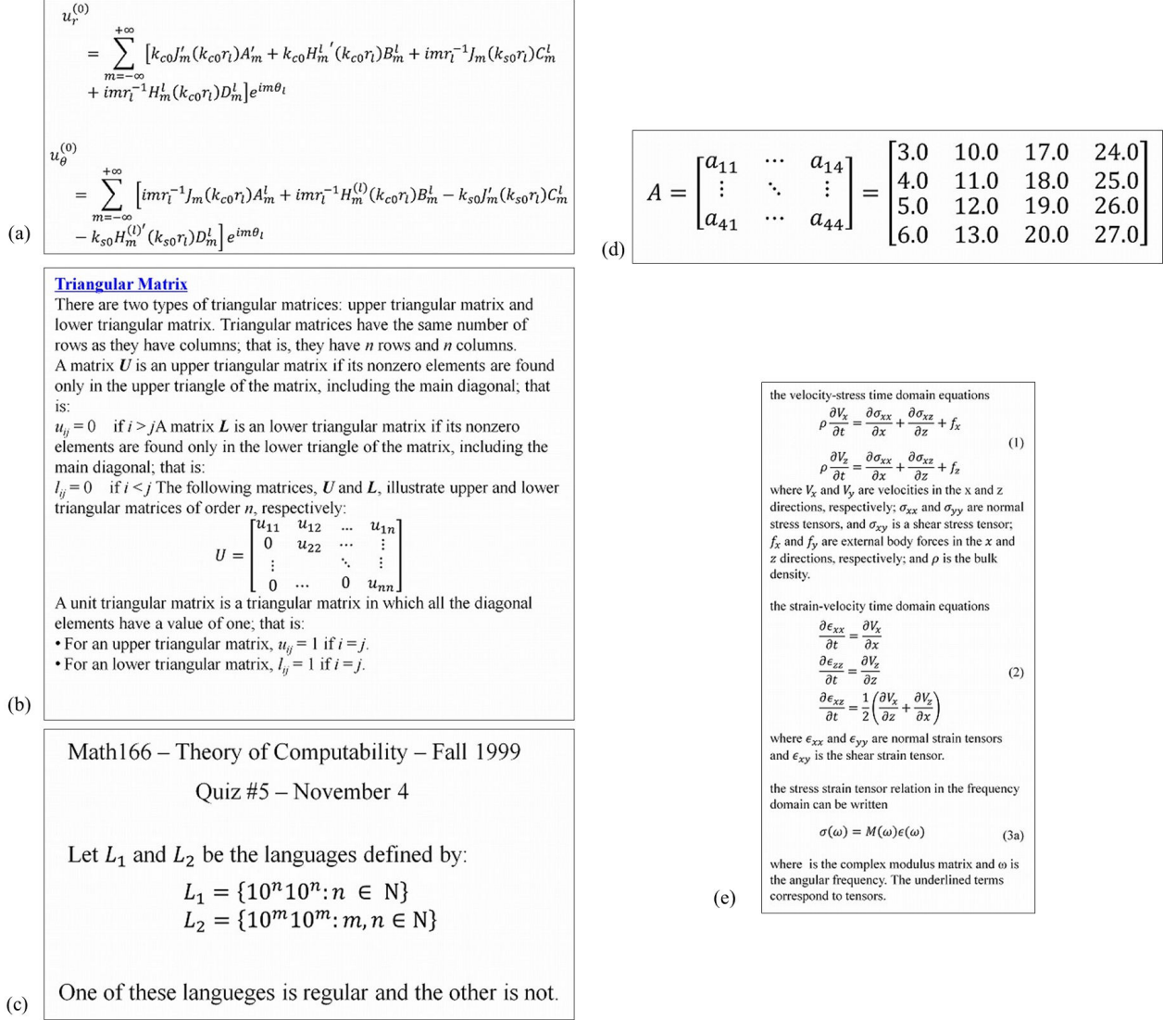


Figure 5: Test images for the comparison experiment.

by adjusting the distance threshold for the grouping to an appropriate value, but usually the optimal threshold is unknown for unknown layout of input images.

The error rate of ± 6 degrees at the largest of the proposed method cannot be considered low. These errors are due to the inaccuracy of the rotation-invariance of SURF. We think the accuracy of the proposed method can be improved by using more suitable features that is to be sought.

5 Conclusion

In this paper, a novel skew estimation method that is applicable to mathematical expressions is proposed. This method is a part-based method. The system detects local parts on an input image and compares them to the preliminarily prepared references.

Whereas conventional methods assume straight text lines in images to estimate the skew, the proposed method does not make such assumption, and therefore it is applicable to mathematical expressions

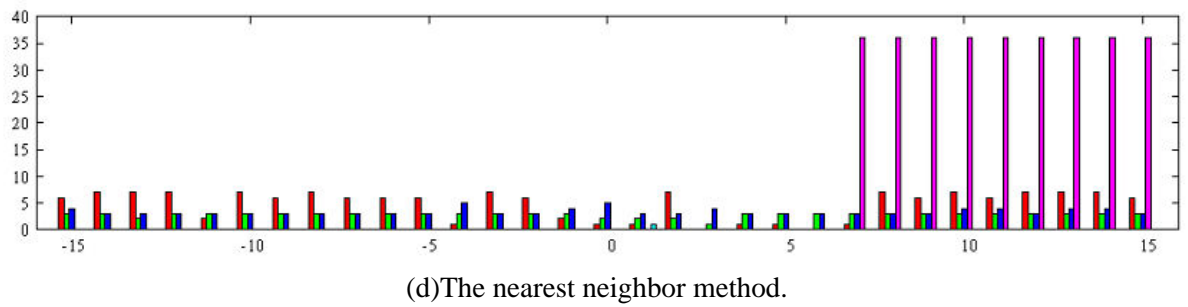
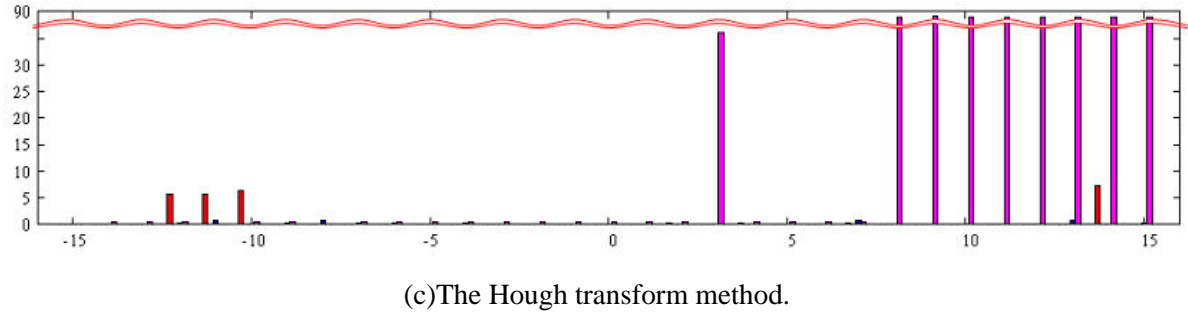
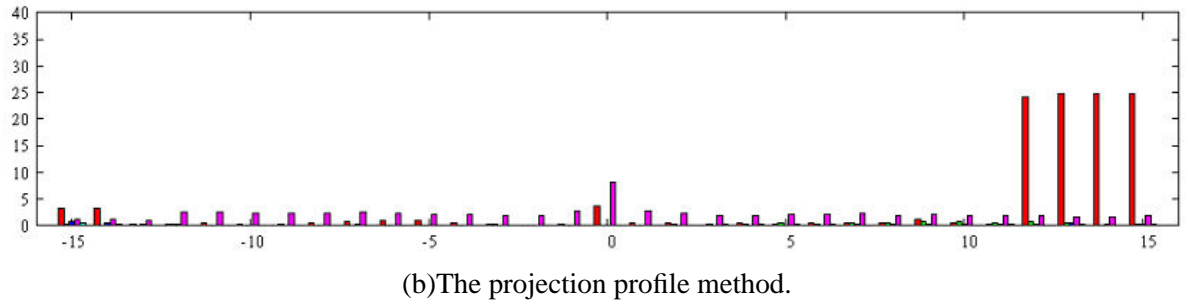
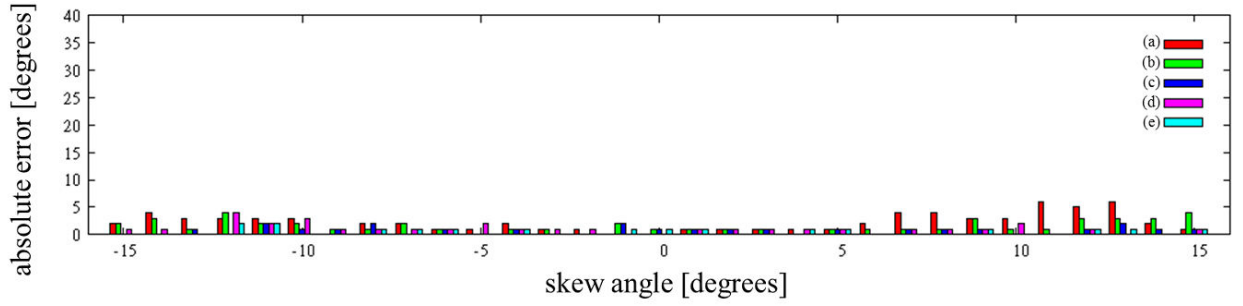


Figure 6: Skew estimation methods comparison.

which often contain ambiguous text lines. The experiments have shown that the proposed method is able to estimated the skew on text images that contain mathematical expressions.

It would be of interest to examine the font dependency of the proposed method in the future. For example one system is possibly able to have only a few font training sets to cover a large variety of fonts (e.g. Times New Roman for Sans-serif and Arial for serif). In the meantime the more suitable feature to

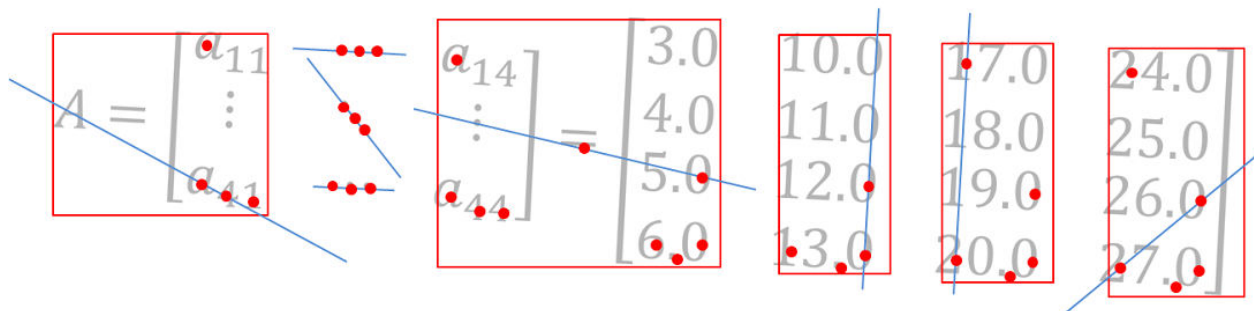


Figure 7: A failed skew estimation made by Hough transform method on test image(d).

describe local parts should be sought in order to further improve the accuracy of the method. It is also challenging to extend the proposed method to the perspective character skew estimation.

References

- [1] J. Hull, "Document Image Skew Detection: Survey and Annotated Bibliography," *Document Analysis Systems II*, World Scientific, pp. 40–64, 1998.
- [2] S. Shiraishi, et al., "A Part-Based Skew Estimation Method," to be published in *Proc. DAS*, 2012
- [3] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," *Proc. ECCV*, 2006.
- [4] W. Postl, "Detection of Linear Oblique Structures and Skew Scan in Digitized Documents," *Proc. ICPR*, pp.687-689, 1986.
- [5] T. Akiyama and N. Hagita, "Automated Entry System for Printed Documents", *Pat. Recognit.*, 23(11), 1990.
- [6] S. N. Srihari and V. Govindaraju, "Analysis of Textual Images Using the Hough Transform," *Mach. Vis. Appl.*, 2, pp.141-153., 1983.
- [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.
instance based method
- [10] S. Uchida, et al., "Skew Estimation by Instances," *Proc. DAS*, pp. 201–208, 2008.
- [11] L. O’Gorman and R. Kasturi, *Document Image Analysis*, IEEE CS Press, 1997.