Rotated Japanese Character Recognition

Yuta Baba, Shiori Sato, Hiroyuki Hase and Shogo Tokai
Graduate School of Engineering, University of Fukui
3-9-1 Bunkyo, Fukui-city, 910-8507, Japan
{y-baba, s-sato}@monju.fuis.u-fukui.ac.jp {haseh,tokai}@u-fukui.ac.jp

***Abstract*—We proposed a rotated character recognition method using eigen-subspace for alpha-numeric characters so far. We first construct an eigen-subspace for each category using the covariance matrix calculated from a sufficient number of rotated character patterns. Next, we can obtain a locus by projecting their rotated characters onto the eigen sub-space and interpolating between their projected points. An unknown character is also projected onto the eigen sub-space of each category. A single projection and multiple projections of the input character image were proposed. Then, the verification is carried out by calculating the distance between the projected points of the unknown character and the locus. Then the multiple projections showed a higher accuracy at low dimensions than a single projection for alphanumeric 62 categories. This time, we applied it for the first class of Japanese Industrial Standard (JIS) Kanji set which includes 2,965 categories. As the result, very high recognition accuracy over 99.8% was achieved by especially multiple projections of the input rotated images.**

***Index Terms*—character recognition, rotated character, eigen-subspace, multiple projections.**

I. Introduction

Some researches on rotated character recognition have been reported so far[1,2]. Recently, we proposed a new scheme on rotated character recognition[4] which was based on the parametric eigen-space method[3]. And also, the other research covering 3D rotation of a character image has been proposed[5]. Both of them targeted 62 categories of alphanumeric letters,. However, these are the same method using eigen-subspace.

In this paper, we apply this method for Japanese characters of First class of JIS Kanji which includes 2,965 categories. We experimented using our method for 2D rotated Kanji characters.

In character recognition research, we generally collect a lot of sample images from the real world, then we develop effective feature extraction and/or recognition algorithm to absorb the intra-class variation, lastly, the recognition performance is evaluated by experiments. An open database of character images may be sometimes used instead of data collection. Printed Kanji image database, ETL2 was opened in 1973[6]. This database consists of 2,184 categories that contains 50,000 data collected from the patent journals and newspapers. But, the number of categories of this database is not enough and collected data is uneven depending on the category and there exists a category which contains only two data. From this meaning, ETL2 is inadequate for our research. Actually, collecting data evenly for all categories is very difficult. In a practical study of character recognition, we must reconsider the recognition result depending on data quality or the method. Deterioration of recognition rate is caused by many factors, i.e. image noise, font, feature extraction, recognition method, and so on. In this paper, we handle noise-free character image data, because those factors affect the system performance and it makes sample image collection easy. Getting these data are difficult in the real world. But, we try to keep a higher image quality as good as possible when we acquire images. Therefore, you can refer to the results of this study as the nearly best result when you could get higher quality image. In this study, the recognition rate depends on only three factors, that is, a lack of information by reducing image size and limited dimensions, and a lot of competing categories.

In our previous work, some simulations[4] using ideal character images were first carried out, then, we got good results close to the simulation result by developing a real-time system[8] with a camera.

In this simulation, we used automatically generated character images. As the results, we could obtain very high accuracy even for any angle of character image of any category of First Class of JIS Kanji.

II. Kanji image generation and Recognition method

In this section, automatic Kanji character image generation is described first, and then the learning and the recognition method are presented[4].

A. Kanji image generation

Kanji images are automatically generated by using Free Type library that is software font engine[7]. By this program, we can get any fonts and any size of bitmap Kanji images.

We generated binary images of First class of JIS Kanji with gothic font and 128×128 pixels. 36 character images rotated by 10 degrees are used for learning process in each category, and these image sizes are changed into 50×50 pixels after extracting the bounding square of a character area. We prepared 51 images for recognition test that are from 7 degrees to 357 degrees rotated by 7 degrees. Therefore 5 images used for the learning are included in test images in each category. In the previous paper[8], between the recognition results experiment using 32×32 binary image for alphanumeric letters and the experimental results using 8×8 pixels with 17 levels, it was no difference in recognition rate. So, in this work, we converted from 50×50 binary image to 8×8 pixels 65 levels image. You may think that a 8×8 pixels image is very coarse for Kanji pattern, but 8×8 pixels images keep their information by the pixel value of 65 levels. This feature representation is one of the easy methods. Computing speed becomes very fast by this resizing. In subsection 3.1, we compare the recognition performance by image size.

B. Learning process

For example, a 50×50 binary image can be described by a 2,500 dimensional vector. The value of a pixel is 0 or 1. a 8×8 image with 65 levels can be described by a 64 dimensional vector. In general, let an image pattern be $f^k_{\theta(i)}$, where k is the category number, $\theta(i)$ is a character angle, that is $\theta(i) = 10 \times i \mid i = 0,1,2,\cdots,35$.

Next, we create the eigen-space using 36 image data with respect to each category. The covariance matrix $\Sigma^{(k)}$ is calculated as follows;

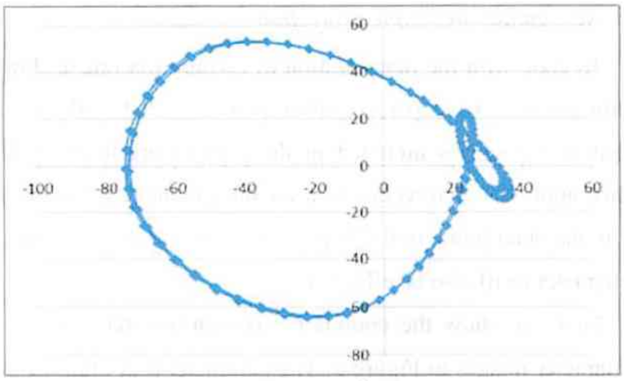
$$\Sigma^{(k)} = E\left[(f^k_{\theta(i)} - m^k)(f^k_{\theta(i)} - m^k)^t\right] \qquad (1)$$

where m^k is the mean vector of the k -th category. The covariance matrix can be obtained through eigen expansion;

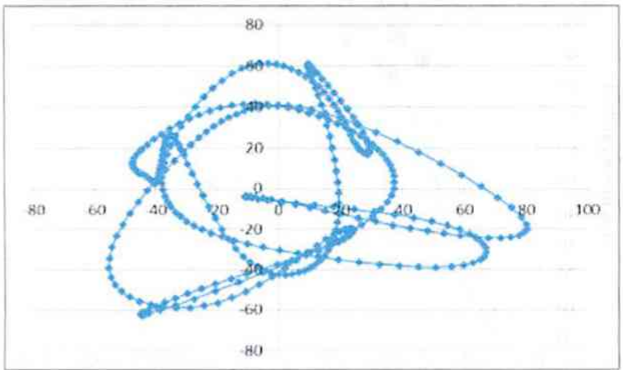
$$\Sigma^{(k)}\phi = \lambda\phi \qquad (2)$$

where, category index k was omitted for λ and ϕ. We obtain at most 35 non-zero eigenvalues because the rank of the covariance matrix is at most 35. Let the eigenvectors corresponding to eigenvalues $\lambda_1, \lambda_2, \cdots, \lambda_{35}$ be $\phi_1, \phi_2, \cdots, \phi_{35}$. Using the first n (<35) eigenvectors, an eigen-subspace $U^{(k)}_n = \{\phi_1, \phi_2, \cdots, \phi_n\}$ can be created.

Then, as projected $f^k_{\theta(i)} (i = 0,1,\cdots,35)$ onto the $U^{(k)}_n$, that is, the projected point $F^k_{\theta(i)}$ is $U^{(k)^t}_n (f^k_{\theta(i)} - m^k)$, a set of the projected points $\{F^k_{\theta(i)}\}$ draws a locus sequentially because the angle changes consecutively. We denote the locus as $L^{(k)}_n$. The locus by 36 points can be interpolated. In this research, 360 points were interpolated by the periodic spline interpolation. The angle of an interpolation point is given integer value by dividing two angles between $F^k_{\theta(i-1)}$ and $F^k_{\theta(i)}$ by ten. Therefore, the precision of angle estimation is at least one degree. We show loci of "末" and "未" on 2D eigen-subspace that their character shapes are similar in Figure 1, but their loci are quite different in this case.



(a) category"末"
Fig.1 Examples of loci(category "末 and 未")



(b) category"未"
Fig.1 Examples of loci(category "末 and 未")

C. Recognition process

(a) Recognition by simple projection

An unknown image x is first projected onto all $U_n^{(k)}$ $(k = 1,2,\cdots,C)$. We will denote the projected point of x as X, that is $X = U_n^{(k)^t}(x - m^k)$. Verification is carried out by finding the shortest distance between X and $L_n^{(k)}$. We represent the shortest distance to the category k as $d^k(X)$. Therefore we can obtain the recognition result k^* as follows;

$$k^* = \arg\min_k \{d^k(X)\} \qquad (3)$$

The angle of the unknown image is given by the angle of closest point on the locus. In this way, we can obtain the recognition result and the character angle of the input image at the same time. We show the recognition scheme by simple projection in Figure 2.

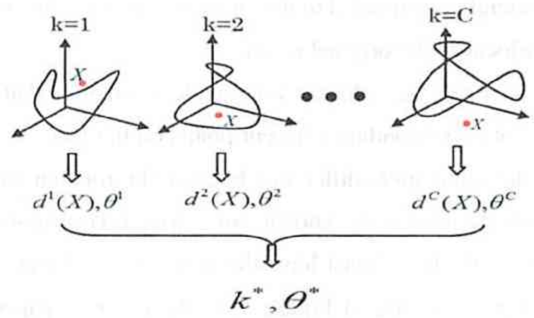


Fig.2 Recognition scheme by simple projection

(b) Recognition by multiple projections

In the previous method, there may be many cases that the projected point is accidentally close to the locus of other category. To prevent this misclassification by accident, we proposed a method that creates multiple images by rotation of an unknown image[4]. These images are projected onto every eigen-subspaces. We obtained higher accuracy for alphanumeric letters. We denote the number of created multiple images including the original image as R. For example, in the case of $R = 3$, two images are created by rotating by 120 degrees and 240 degrees. In our previous work, we changed R from 1 to 5, as the result, R=3 or 5 showed higher performance than R=2 or 4. In addition, recognition by simple projection is the case of $R = 1$.

Figure 3 shows a recognition scheme of multiple projections in case of $R = 3$, the distance of the category is defined as the average distance among three distances.

$$k^* = \arg\min_k [E_j\{d_j^k(X)\}] \qquad (4)$$

Where, $E\{\cdot\}$ is average operation.

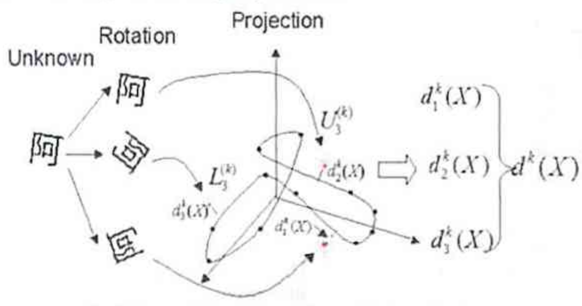


Fig.3 Recognition scheme by multiple projections

The angle is estimated to the angle of corresponding point on the locus to the original image.

In addition, we adopt a reasonable restriction that the angles of corresponding adjacent points on the locus should have the same angle difference because the rotation angles of multiple images are known. So, adjacent corresponding points on the locus must have the same angle difference of 120 degrees in case of Figure 3. In the previous paper[4], we did not use this restriction. This restriction is one of ingenuities for handling a lot of categories.

III. EXPERIMENTAL RESULTS AND CONSIDERATIONS

A. Recognition performance for size of character image

In recognition experiment by simple projection, Figure 4 shows recognition rates by the simple projection using the first n dimensions for 50×50 binary images and 8 × 8 65 levels images. The recognition rate using the first 35 dimensions is 99.80% for 50×50 character images, and 99.74%(150,826 correct per 151,215 samples) for 8×8 character images. Both results are very high.

The accuracy for 8×8 character images is lower than the one of 50×50 character images at low dimensions. But, the deterioration of recognition rate is not seen at high dimensions. This is conceivable that the loss of the information is suppressed by 65 levels of 8×8 character image. Since we will show experimental results using 8×8 character images hereafter.

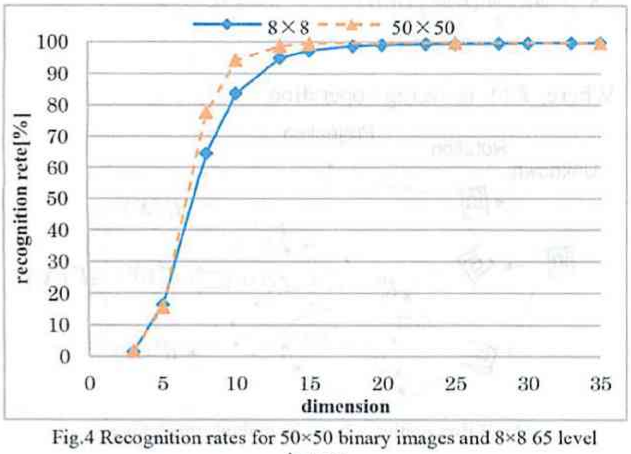


Fig.4 Recognition rates for 50×50 binary images and 8×8 65 level images

B. Recognition by simple projection

To cope with the deterioration of recognition rate at low dimensions, two-step recognition process will be effective, that is, categories included in the p-place are nominated, then apply detailed recognition for the candidate categories. On the other hand, multiple projections by rotated unknown characters will also be effective.

First, we show the cumulative recognition rate for 8×8 character images in Figure 5. The parameter p in Figure 5 is the number of best candidates.

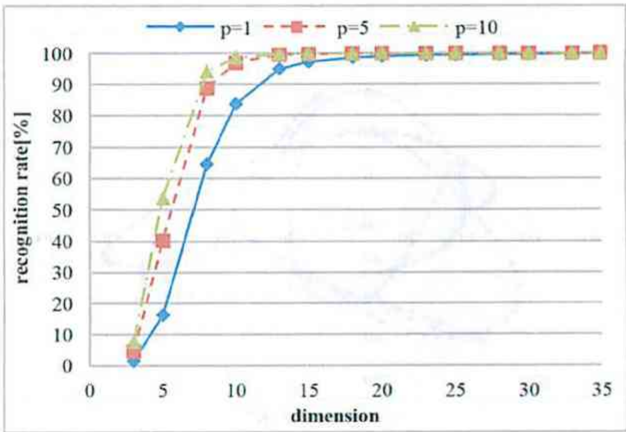


Fig.5 Cumulative recognition rate

The cumulative recognition rate at 10 dimensions for p=10 is 98.66%. In addition, all true categories were included in 23 best candidates with the first 35 dimensions. Further classification for the best candidates will be effective by using high resolution image, as 50×50 images already showed high recognition rate in low dimensions in Figure 4.

C. Recognition by multiple projections

Next, we show the recognition rate by multiple projections of $R=3$ in Figure 6. You can see that the recognition performance at low dimensions is quite higher than that of $R=1$.

The recognition rate of R=3 at the first 10 dimensions is 99.66%, and 99.99%(20 errors in 151,215 test samples) at the first 35 dimensions.

We can cover a deterioration of the recognition rate at low dimensions by the two-stage classification and by the multiple projections. In recognition by the multiple projections, we got high recognition rate at low dimensions because contingency in simple projection can be suppressed.

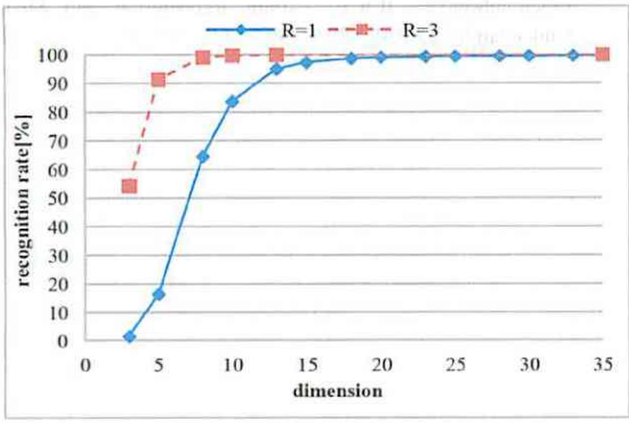


Fig.6 Recognition rate by multiple projections(R=3)

We show a histogram of angle difference between true angle and estimated one in Figure 7. This graph is depicted using truly recognized samples and estimated angles are restricted within ±7 degrees, and misclassified sample are excluded. Furthermore, 26 samples of category "一" are excluded because their estimated angles were upside down. All samples correctly recognized except "一" are within ±3 degrees. The most are high precision within ±1 degrees.

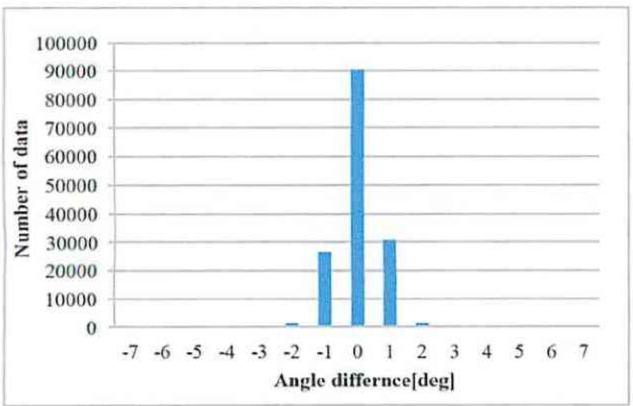


Fig.7 Precision of estimated angle errors

From these results, you can see that interpolated angles of 360 points are nearly correct. In recognition by simple projection and recognition by multiple projections($R=3$), examples of misclassification using the first 35 dimensions are shown in Table 1. Misclassified categories are very similar to the input categories. But, their estimated angles are within ±1 degrees though they are misclassified. In addition, in misclassified samples by the multiple projections(R=3), their true categories were included in the best two candidates.

TABLE I. EXAMPLES OF MISCLASSIFICATION

Input Result	R=1	R=3
晋 ⇒ 普	24	14
震 ⇒ 霊	11	0
鳳 ⇒ 風	11	0
漬 ⇒ 潰	9	3
漠 ⇒ 漢	9	1
島 ⇒ 鳥	8	0
廷 ⇒ 延	7	0
奥 ⇒ 臭	6	0
筒 ⇒ 箭	6	0
諭 ⇒ 論	6	0
others	292	2

D. Computation time

The specification of the system used for the simuration is as follows;

CPU : Intel Xeon(6core/12thread),Memory : 48GB.

In the recognition process, all categories are divided into 15 threads for parallel computation.

(a) The case of 50×50 binary images using 35 dimensions

The computation time per a character is 2.26[s]

Memory for the dictionary : 2.43[GB]

(b) The case of 8×8 65 levels image using 35 dimensions

The computation time per a character is 0.20[s]

Memory for the dictionary : 353.5[MB]

(c) The case of 8×8 65 levels image using the first 10 dimensions by multiple projections($R=3$)

The computation time per a character is 0.15[s]

IV. SUMMARY

We proposed a rotated character recognition method using eigen-subspace for alpha-numeric characters before, In this paper, we applied it for the first class of Japanese Industrial Standard (JIS) Kanji set which includes 2,965 categories. As the result, we have obtained very high recognition rate. At first, we showed that the recognition rates at 35 dimensions for different sizes of image are almost same. But, it decreases a little for small image size at low dimensions. Next, we experimented by two recognition methods that were the simple projections using only input image and the multiple projections using rotated images created by rotation of the input image. As the results, we have got very higher recognition rate than single projection. The recognition rate is 99.74% for simple projection using 35 dimensions, and 99.99% for multiple projections($R=3$)using 35 dimensions. Especially, the multiple projections showed high performance at low dimensions. In the future, we will consider the influence of the noise on a real image that exerts to the recognition rate. Furthermore, we try to absorb variations of real image by effective feature extraction. In addition, we want to develop a real time system.

REFERENCES

[1] Q.Xie, A.Kobayashi, "A Construction of Pattern Recognition System Invariant of Translation, Scale-Change and Rotation Transformation of Patterns(in Japanese), " Trans. of The Society of Instrument and Control Engineers, Vol.27,No.10, pp.1167-1174 (1991).

[2] S.Sato, S.Miyake and H.Aso, "Evaluation of Two Neocognitron-type Models for Recognition of Rotated Patterns," ICONIP 2000, WBP-04,pp295-299 (2000).

[3] H.Murase, S. K. Nayar, "3D Object Recognition from Appearance-Parametric Eigenspace Method-(in Japanese)," Trans. of IEICE, Vol.J77-D-II,No.11, pp.2179-2187 (1994).

[4] Toshiyuki Sinokawa, Hiroyuki Hase, Hiroshi Kakutani, Masaaki Yoneda, "Recognition of Rotated Characters by the Parametric Eigen-space Method", IIEEJ, vol.33,no.6,pp.1123-1131(2004.11)(in Japanese).

[5] Ryo Narita, Wataru Ohyama, Tetsushi, Wakabayashi, Fumitaka Kimura , "3D rotation invariant character recognition", MIRU2011, OSI-4: 26-32, (2011)(in Japanese).

[6] http://projects.itri.aist.go.jp/etlcdb/etln/etl2/etl2.htm

[7] Shukun NING, Hiroyuki HASE, Shogo TOKAI, "Kanji Character Image Generation for Character Recognition", 2012 Joint Conference of Hokuriku Chapters of Electrical Societies, F-56, 2012.

[8] Masafumi Yamamoto, Shiori Ssato, Hiroyuki Hase, Shogo Tokai, "Real-time Rotated Character Recognition using Eigen-subspace", IEICE, Pattern Recognition and Media Understanding, PRMU2010-246,55-60,2011-03.