

Global Shape Normalization for Handwritten Chinese Character Recognition: A New Method

Cheng-Lin Liu and Katsumi Marukawa
Central Research Laboratory, Hitachi, Ltd.
1-280 Higashi-koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
{liucl,marukawa}@crl.hitachi.co.jp

Abstract

Nonlinear normalization (NLN) based on line density equalization has been widely used in handwritten Chinese character recognition (HCCR). Our previous results showed that global transformation methods, including moment normalization and a newly proposed bi-moment method, generate smooth normalized shapes at lower computation effort while yielding comparable recognition accuracies. This paper proposes a new global transformation method, named modified centroid-boundary alignment (MCBA) method, for HCCR. The previous CBA method can efficiently correct the skewness of centroid by quadratic curve fitting but fails to adjust the inner density. The MCBA method adds a simple trigonometric (sine) function onto quadratic function to adjust the inner density. The amplitude of the sine wave is estimated from the centroids of half images. Experiments on the ETL9B and JEITA-HP databases show that the MCBA method yields comparably high accuracies to the NLN and bi-moment methods and shows complementarity.

1. Introduction

The performance of handwritten Chinese character recognition (HCCR) is largely dependent on shape normalization, and the nonlinear normalization (NLN) based on line density equalization [1, 2] has been widely adopted to fulfill this task. The line density-based NLN methods, however, are complicated in computation and the normalized shapes are not smooth due to the local transformation nature. On the contrary, global transformation methods estimate very few parameters efficiently from global shape features and generate smooth normalized shapes. For examples, the one-dimensional moment normalization method (centroid alignment without rotation or shearing, as a simplification of Casey's method [3]) was shown to perform comparably well to NLN, and a newly proposed bi-moment method, performs slightly better than the moment method

[4]. Another method that aligns both centroid and character boundary, called centr-bound (abbreviation of centroid-boundary alignment, CBA) method, performs fairly well.

The superiority of CBA, moment and bi-moment methods indicates that aligning the centroid of character image to the geometric center is efficient to reduce the within-class shape variation. The moment and bi-moment methods further regulate the extent of central stroke area by re-setting the boundaries of character area. The CBA method aligns the centroid and the physical boundaries (the bounds of projections of stroke area) using quadratic curve fitting. The quadratic curve is not able to deflect the inner density, however. To overcome this problem, we propose an improved CBA method, called modified CBA (MCBA). To adjust the inner density after quadratic curve fitting, we transform the coordinates of pixels using a simple trigonometric function, the sine function of one period. The amplitude of the sine wave is estimated automatically from the extent of the central area of character. This add-on transformation brings on very little overhead. We will show that the MCBA method yields comparable recognition accuracies to the NLN and bi-moment methods.

Unlike the MCBA method, the moment and bi-moment methods do not align the physical boundaries, and so, the re-set boundaries may cut off strokes or leave blank margins in the normalized plane. Due to the different normalized shapes, these methods will generate different recognition results on some specific character images though their recognition accuracies are comparable statistically. Hence, the fusion of their recognition results may yield even higher performance.

We have also applied the MCBA method to fitting the line density projections of NLN. The resulting method, called line density projection fitting (LDPF), generates smoother normalized images than the original NLN and yields similar recognition accuracies.

As to other normalization methods, the cosine function transformation of Guo et al. [5] is also a global transformation method. Unlike that we estimate the transformation parameters from the character shape features, they trans-

formed the character image into multiple normalized images using pre-specified parameters and then selected the best performing one. Jin et al. extended the trigonometric transformation functions to generate more flexible deformations but the parameters remain artificially specified [6]. On the other hand, the class-dependent normalization [7, 8] and the two-dimensional extension of NLN [9] are very computationally expensive.

The rest of this paper is organized as follows. Section 2 describes the MCBA method; Section 3 presents the experimental results, and Section 4 provides concluding remarks.

2. Modified CBA Method

The insufficiency of the CBA method lies in the inability of correcting the imbalance of inner/outer density. The moment and bi-moment methods adjust the inner density by re-setting the character boundaries. The CBA method fails to do this because the coordinate mapping function is confined to quadratic and the physical boundaries are taken. In the modified CBA (MCBA) method, on quadratic coordinate mapping, we further adjust the inner density using a sine function. The period of the sine function is fixed and its amplitude is estimated from the extent of the central area of character image.

For convenience of illustration, we assume that the coordinates of both the input image and the normalized image have been re-scaled to $[0, 1]$. The CBA method then aligns three points $\{0, x_c, 1\}$ or $\{0, y_c, 1\}$ (the centroid coordinates x_c and y_c have been re-scaled as well) to $\{0, 0.5, 1\}$. In the following, we describe the coordinate mapping functions of x -axis, while those of y -axis can be obtained by simply substituting x with y .

Because the bounds of coordinates are scaled to 0 and 1, the quadratic function for centroid alignment is simplified to $ax^2 + bx$. Fig. 1(a) shows two curves of quadratic functions. The curve "A" aligns a centroid with $x_c < 0.5$ to the center 0.5 and effects in expanding the character image in the left and compressing in the right, while the curve "B" aligns a centroid with $x_c > 0.5$ and effects in compressing the character image in the left and expanding in the right.

Denote the coordinate after centroid alignment by $z = ax^2 + bx$, we further adjust (stretch or compress) the inner density using a sine function of one period. The coordinate mapping function is

$$x' = z + \eta \sin(2\pi z), \quad z \in [0, 1], \quad (1)$$

where η is the amplitude of the sine wave. Fig. 1(b) shows two curves of functions for adjusting inner density. The curve "C" has $\eta < 0$ and effects in stretching the inner area, while the curve "D" has $\eta > 0$ and effects in compressing the inner area.

The combination of quadratic function and sine function results in a coordinate mapping function that can both align

the centroid and adjust the inner density:

$$x' = ax^2 + bx + \eta \sin(2\pi(ax^2 + bx)). \quad (2)$$

Fig. 1(c) shows four curves of combined mapping functions. The curve "A*C" combines the effects of compressing the right and stretching the inner, the curve "A*D" combines the effects of expanding the left and compressing the inner, the curve "B*C" combines the effects of compressing the left and stretching the inner, and the curve "B*D" combines the effect of expanding the right and compressing the inner.

Considering that the deformations of handwritten characters mainly appear to be the skewness of centroid and the imbalance of inner/outer density, the combination of quadratic function and sine function is able to correct very well these deformations, and consequently, lead to high recognition performance.

2.1 Parameter estimation

The amplitude of the sine wave is the only parameter of coordinate mapping function for adjusting inner density. It can be estimated from the extent of the central area of character image. We truncate the central area with the local centroids of the half images left/right (or upper/lower) to the global centroid (x_c, y_c) . Denote the x -coordinate of the centroid of the left half by x_1 and that of the right half by x_2 (Fig. 2). On centroid alignment with quadratic function, they are transformed to values $z_1 = ax_1^2 + bx_1$ and $z_2 = ax_2^2 + bx_2$. The extent of the central area is then

$$s_x = z_2 - z_1 = ax_2^2 + bx_2 - ax_1^2 - bx_1. \quad (3)$$

The bounds of the central area is re-set to be equally distant from the aligned centroid: $0.5 - s_x/2$ and $0.5 + s_x/2$. The sine function aims to map these two bounds to coordinates 0.25 and 0.75 in the normalized plane. This gives

$$0.5 - \frac{s_x}{2} + \eta \sin(2\pi(0.5 - \frac{s_x}{2})) = 0.25$$

and

$$0.5 + \frac{s_x}{2} + \eta \sin(2\pi(0.5 + \frac{s_x}{2})) = 0.75.$$

These two equations are equivalent and result in

$$\eta = \frac{s_x/2 - 0.25}{\sin(\pi s_x)}. \quad (4)$$

From (4), we can see that when $s_x < 0.5$, $\eta < 0$. This corresponds to the curve "C" of Fig. 1 and indicates that when the central area is dense (focusing toward the centroid), the sine function will stretch the inner area. When $s_x > 0.5$, $\eta < 0$ corresponds to the curve "D" of Fig. 1 and indicates that when the central area extends outward, the sine function will compress the inner area.

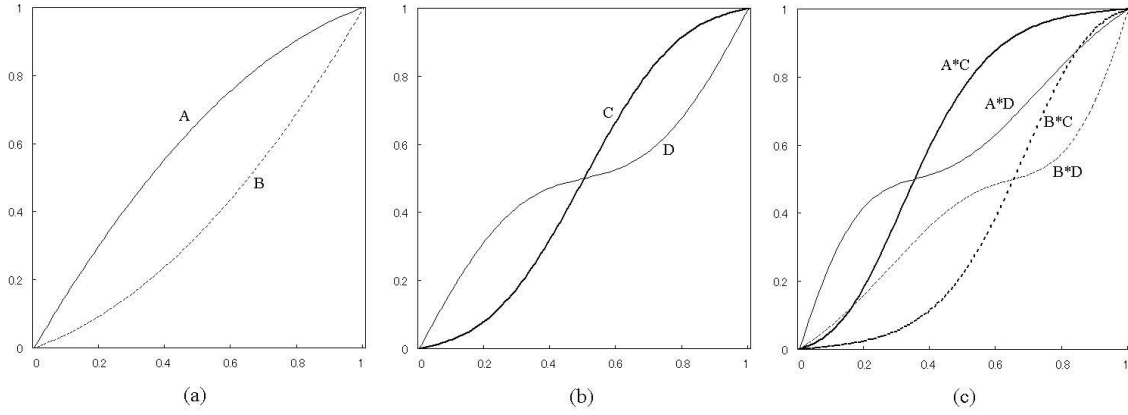


Figure 1. Curves of coordinate mapping: (a) quadratic curve fitting for centroid alignment; (b) sine functions for adjusting inner density; (c) combination of quadratic and sine functions.

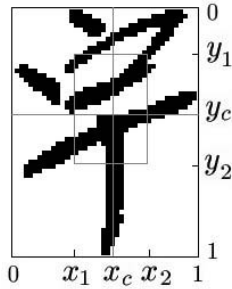


Figure 2. Centroid and local centroids of half images. The central area is enclosed by the gray rectangle defined by the local centroid coordinates of four half images.

The amplitude of the sine wave should be constrained to preserve the monotonicity of coordinates. The mapped coordinates x' and y' must be increasing with x or y :

$$\frac{dx'}{dx} \geq 0.$$

Substituting (2) gives

$$(2ax + b)[1 + 2\pi\eta \sin(ax^2 + bx)] \geq 0.$$

$2ax + b \geq 0$ has been satisfied by the monotonicity constraint of $ax^2 + bx$, while $1 + 2\pi\eta \sin(ax^2 + bx) \geq 0$ is satisfied by

$$-\frac{1}{2\pi} \leq \eta \leq \frac{1}{2\pi}. \quad (5)$$

When the value of η computed by (4) is beyond this range, we enforce it to be a marginal value $-1/2\pi$ or $1/2\pi$, which corresponds to $s_x = 0.265$ or $s_x = 0.735$.

2.2 Line density projection fitting (LDPF)

The proposed MCBA method can also be used to fit the projections (onto x/y axis) of line densities of NLN. The line density projections are fitted into smooth coordinate mapping functions such that the generated normalized image has smoother stroke shapes. In the NLN method, the horizontal line densities are projected onto x -axis. The projection values are normalized to unity of sum and then accumulated to give normalized coordinates in $[0, 1]$. For fitting the normalized coordinates using MCBA, we pick up the coordinates (of input image) x_1 , x_c , and x_2 , that correspond to normalized coordinates (from accumulation) 0.25, 0.5, and 0.75, respectively. The coordinate x_c is then aligned to 0.5 using a quadratic function. x_1 and x_2 are used to compute the extent of central area and estimate the amplitude of a sine function. Finally, the input image is transformed according to the fitted coordinate mapping functions instead of the line density projections.

2.3 Implementation notes

In our experiments, the normalized image plane was set to 64×64 pixels. The aspect ratio of the input image is partially preserved using the so-called aspect ratio adaptive normalization (ARAN) strategy [10]. In ARAN, the aspect ratio R_2 of the normalized image is a continuous function of the aspect ratio R_1 of the input image. We adopt the aspect ratio function

$$R_2 = \sqrt{\sin\left(\frac{\pi}{2} R_1\right)}.$$

R_1 is calculated by

$$\begin{cases} R_1 = W_1/H_1, & \text{if } W_1 < H_1 \\ R_1 = H_1/W_1, & \text{otherwise} \end{cases}$$

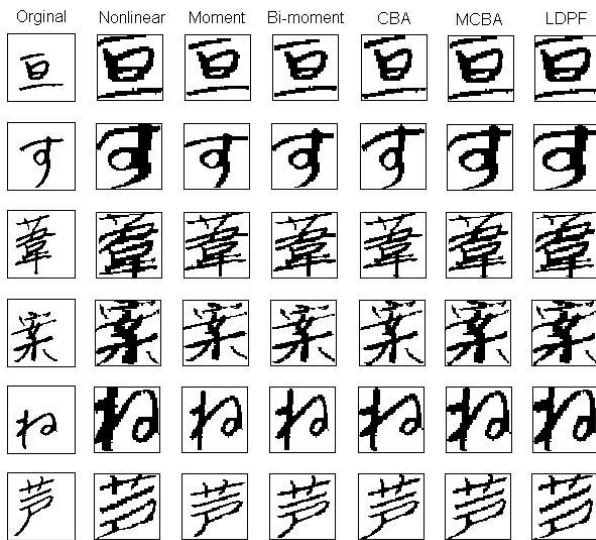


Figure 3. Examples of character shape normalization.

where W_1 and H_1 are the width and height of the input image. For linear normalization, line density-based NLN, CBA and MCBA methods, the width and height are taken as the extents of stroke projections. For moment and bi-moment methods, W_1 and H_1 are re-set according to second-order moments [4]. If the input image is vertically elongated, then in the normalized plane, the vertical dimension is filled and the horizontal dimension is centered and scaled according to the aspect ratio; otherwise the horizontal dimension is filled and the vertical dimension is centered and scaled.

Fig. 3 shows some examples of normalization by six methods: NLN, moment, bi-moment, CBA, MCBA, and LDPF. In the figure, the leftmost column shows the input images, and the other columns shows the six normalized images. We can see that the MCBA method corrects both the skewness of centroid and the imbalance of inner/outer density. In the first four rows, the MCBA method stretches the central area of the input image horizontally. The input image of the fifth row is compressed horizontally in the central area. The input images of the second and last rows are apparently skewed with respect to the centroid. We can also see that the global transformation methods generate smoother normalized shapes than NLN. The normalized image generated by LDPF is very similar to that of NLN but has smoother stroke shapes.

3. Experimental Results

We evaluated the performance of the normalization methods in HCCR on two large databases: ETL9B and JEITA-HP. The ETL9B database was collected and released

by the Electro-Technical Laboratory (ETL) of Japan (currently named the National Institute of Advanced Industrial Science and Technology (AIST)). It contains the binary images of 3,036 characters (including 71 hiragana and 2,965 JIS level-1 Kanji), 200 images per category. We used the first 20 images and the last 20 images of each class for testing, and the rest 160 images for learning classifier parameters.

The JEITA-HP database was originally collected by Hewlett-Packard Japan and later released by JEITA (Japan Electronics and Information Technology Association). It contains the character images of 580 writers, including 480 writers (A0–492 with 13 numbers absent) in DATASET-A and 100 writers (B0–99) in DATASET-B. Experimental results on JEITA-HP database have been reported by Kawatani et al. [11] but their specification of training and test data is not clear. To compare the results of JEITA-HP with those of ETL9B, we have figured out the 3,036 categories of ETL9B from JEITA-HP [4]. Let us refer to the 3,036 images of one writer as a set. We used the first 400 sets of DATASET-A and the first 80 sets of DATASET-B for training, and the rest 80 sets of DATASET-A and 20 sets of DATASET-B for testing. The total numbers of images of training set and test set are 1,441,906 and 303,334, respectively.

From a character image, we extracted chaincode direction features using normalization-base feature extraction (NBFE) or normalization-cooperated feature extraction (NCFE). By NBFE, the normalized image is generated, then the contour pixels are assigned to four orientation planes, from each plane feature measurements are computed by blurring (Gaussian filtering and sampling) [12]. While by NCFE, the contour pixels of the input binary image are assigned to orientation planes incorporating coordinate mapping [13]. We have improved the performance of NCFE by generating continuous orientation planes instead of discrete ones [14]. We also extracted gradient direction feature (*grd-g*) from gray-scale normalized image (transformed from binary character image) [14]. The size of normalized image and orientation planes was set to 64×64 pixels, and from each orientation plane, we extracted $8 \times 8 = 64$ measurements. Variable transformation was imposed on each measurement to improve the Gaussianity of feature distribution [15, 16]. We set the power to 0.5 without attempt to optimize it.

We present the results of two classifiers, namely, Euclidean distance to class mean and modified quadratic discriminant function (MQDF2) [17], both on reduced feature vector by Fisher linear discriminant analysis [15]. The dimensionality of feature vector was reduced to 160 without loss of classification accuracy. The MQDF2 was proposed by Kimura et al. to reduce the storage and computation of ordinary QDF and to improve the classification performance [17]. In MQDF2, the covariance matrix of each class is regularized by replacing the minor eigenvalues with

a constant. We set the number of principal eigenvectors to 50. The class-independent constant minor eigenvalue was heuristically set to be proportional to the average feature variance and the multiplier was selected to maximize the accuracy on a holdout set randomly extracted from the training set. The parameters of MQDF2 were then re-estimated on the whole training set. We speeded up the classification of MQDF2 by selecting 100 candidate classes using Euclidean distance. The MQDF2 was then computed on the candidate classes only.

The recognition rates on two databases are shown in Table 1 and Table 2. For line density-based NLN, we give the results of two line density definitions, namely, "NLN-T" of Tsukumo et al. [1] and "NLN-Y" of Yamada et al. [2]. We have given slight modifications to them. Tsukumo et al. did not specify the line density computation of marginal and stroke areas empirically so as to achieve high recognition accuracy. Yamada et al. elaborated the density computation for all configurations. For improving the recognition performance, we adopted the modification of Yoshida et al. [18], that re-defined the line interval of marginal area and the unification of horizontal and vertical intervals. The LDPF method fits the line densities computed by the NLN-T.

Table 1. Recognition rates (%) on ETL9B test set.

Classify	Euclidean			MQDF2		
	NBFE	NCFE	grd-g	NBFE	NCFE	grd-g
NLN-T	96.59	97.06	97.06	98.82	99.03	98.98
NLN-Y	96.44	96.75	96.83	98.76	98.88	98.89
Moment	96.84	97.06	97.12	98.81	98.95	98.99
Bi-mom	96.90	97.10	97.18	98.85	98.96	99.01
CBA	96.22	96.45	96.57	98.64	98.74	98.79
MCBA	96.59	96.89	96.97	98.76	98.93	98.94
LDPF	96.39	96.78	96.85	98.73	98.94	98.94

Since the accuracies of MQDF2 are significantly higher than those of Euclidean distance, we focus on the accuracies of MQDF2 for comparing the normalization methods. On both ETL9B and JEITA-HP, we can see that five normalization methods, namely, NLN-T, moment, bi-moment, MCBA, and LDPF, yield comparable accuracies with the two good features NCFE and *grd-g*. When extracting chain-code feature from normalized images (NBFE), the NLN and LDPF perform inferiorly on JEITA-HP, probably because the normalized images are not smooth though we performed smoothing after normalization. For all the feature types and classifiers, the NLN-T outperforms NLN-Y owing to our modifications to the line density computation. The LDPF method yields similar accuracies to the NLN-T.

Table 2. Recognition rates (%) on JEITA-HP test set.

Classify	Euclidean			MQDF2		
	NBFE	NCFE	grd-g	NBFE	NCFE	grd-g
NLN-T	93.18	94.05	93.88	97.57	97.93	97.83
NLN-Y	92.95	93.51	93.48	97.47	97.76	97.68
Moment	93.68	94.14	94.21	97.70	97.93	97.91
Bi-mom	93.76	94.25	94.28	97.76	98.00	97.98
CBA	92.72	93.25	93.33	97.49	97.76	97.73
MCBA	93.52	94.05	94.14	97.74	98.00	97.97
LDPF	93.01	93.66	93.68	97.53	97.88	97.80

As we have analyzed in the above, the normalization methods generate different normalized shapes and may give different recognition results on specific character images though their recognition rates are comparable statistically. To justify the complementariness of these methods, we show some character samples that are misclassified by MQDF2 with NCFE using NLN-T, bi-moment, or MCBA normalization. Some samples that were given different results by three normalization methods are shown in Fig. 4. We can see that if we re-classify according to the votes of three methods, we can get correct classification to many samples that are misclassified by individual methods.



Figure 4. Character images misclassified by MQDF2 using three normalization methods: NLN, bi-moment, and MCBA.

To evaluate the computation complexity of the methods, we profiled the CPU times of coordinate mapping by seven normalization methods. On an input character image, the CPU time is counted until the transformed coordinates are

computed. It does not cover either the normalized image generation or the feature extraction procedure. We implemented the experiments on Pentium4-1.90GHz and averaged the CPU time over the samples of JEITA-HP test set. The CPU times are shown in Table 3. We can see that the global transformation methods (moment, bi-moment, CBA and MCBA) are less computationally expensive than line density-based NLN. The MCBA method adds little additional computation overhead to CBA.

Table 3. Processing time of normalization on JEITA-HP test set.

Method	CPU (ms)
NLN-T	0.271
NLN-Y	0.387
Moment	0.049
Bi-moment	0.051
CBA	0.053
MCBA	0.067
LDPF	0.283

4. Conclusion

The proposed global normalization method, MCBA method, combines centroid alignment and inner area compressing/stretching, and hence is able to correct both the skewness with respect to the centroid and the imbalance of inner/outer density. In experiments of HCCR on large databases, the MCBA method was shown to yield comparably high recognition accuracies to previous best performing methods and be complementary on specific samples. The global transformation methods are readily applicable to on-line trajectories and gray-scale character images since the parameters are estimated from one-dimensional projections, unlike the line density-based NLN that must be computed on two-dimensional images.

References

- [1] J. Tsukumo, H. Tanaka, Classification of handprinted Chinese characters using non-linear normalization and correlation methods, *Proc. 9th ICPR*, Roma, Italy, 1988, pp.168-171.
- [2] H. Yamada, K. Yamamoto, T. Saito, A nonlinear normalization method for hanprinted Kanji character recognition—line density equalization, *Pattern Recognition*, 23(9): 1023-1029, 1990.
- [3] R.G. Casey, Moment normalization of handprinted character, *IBM J. Res. Dev.*, 14: 548-557, 1970.
- [4] C.-L. Liu, H. Sako, H. Fujisawa, Handwritten Chinese character recognition: alternatives to nonlinear normalization, *Proc. 7th ICDAR*, Edinburgh, Scotland, 2003, pp.524-528.
- [5] J. Guo, N. Sun, Y. Nemoto, M. Kimura, H. Echigo, R. Sato, Recognition of handwritten characters using pattern transformation method with cosine function, *Trans. IEICE Japan*, J76-D2(4): 835-842, 1993 (in Japanese).
- [6] L. Jin, J. Huang, J. Yin, Q. He, Deformation transformation for handwritten Chinese character shape correction, *Advances in Multimodal Interfaces—ICMI2000*, LNCS Vol. 1948, T. Tan, Y. Shi, and W. Gao (Eds), Springer, 2000, pp.450-457.
- [7] T. Wakahara, K. Odaka, Adaptive normalization of handwritten characters using global/local affine transformation, *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(12): 1332-1441, 1998.
- [8] M. Nakagawa, T. Yanagida, T. Nagasaki, An off-line character recognition method employing model-dependent pattern normalization by an elastic membrane model, *Proc. 5th IC-DAR*, Bangalore, India, 1999, pp.495-498.
- [9] T. Horiuchi, R. Haruki, H. Yamada, K. Yamamoto, Two-dimensional extension of nonlinear normalization method using line density for character recognition, *Proc. 4th IC-DAR*, Ulm, Germany, 1997, pp.511-514.
- [10] C.-L. Liu, M. Koga, H. Sako, H. Fujisawa, Aspect ratio adaptive normalization for handwritten character recognition, *Advances in Multimodal Interfaces—ICMI2000*, LNCS Vol. 1948, T. Tan, Y. Shi, and W. Gao (Eds), Springer, 2000, pp.418-425.
- [11] T. Kawatani, H. Shimizu, Handwritten Kanji recognition with the LDA method, *Proc. 14th ICPR*, Brisbane, Australia, 1998, Vol.2, pp.1031-1035.
- [12] C.-L. Liu, Y.-J. Liu, R.-W. Dai, Preprocessing and statistical/structural feature extraction for handwritten numeral recognition, *Progress of Handwriting Recognition*, A.C. Downton and S. Impedovo (Eds.), World Scientific, 1997, pp.161-168.
- [13] M. Hamanaka, K. Yamada, J. Tsukumo, Normalization-cooperated feature extraction method for handprinted Kanji character recognition, *Proc. 3rd IWFHR*, Buffalo, NY, 1993, pp.343-348.
- [14] C.-L. Liu, K. Nakashima, H. Sako, H. Fujisawa, Handwritten digit recognition: investigation of normalization and feature extraction techniques, *Pattern Recognition*, 37(2): 265-279, 2004.
- [15] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed., Academic Press, 1990.
- [16] F. Kimura, T. Wakabayashi, S. Tsuruoka, Y. Miyake, Improvement of handwritten Japanese character recognition using weighted direction code histogram, *Pattern Recognition*, 30(8): 1329-1337, 1997.
- [17] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake, Modified quadratic discriminant functions and the application to Chinese character recognition, *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(1): 149-153, 1987.
- [18] A. Yoshida, Y. Hongo, Handprinted Kanji recognition by the background feature pattern matching method with a nonlinear normalization, *IEICE Technical Report*, PRU92-34 (1992-09) (in Japanese).