Handwritten Chinese Character Recognition: Alternatives to Nonlinear Normalization

Cheng-Lin Liu, Hiroshi Sako, and Hiromichi Fujisawa
Central Research Laboratory, Hitachi, Ltd.
1-280 Higashi-koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
E-mail: {liucl,sakou,fujisawa}@crl.hitachi.co.jp

Abstract

Nonlinear normalization (NLN) by line density equalization has been popularly used in handwritten Chinese character recognition (HCCR). To overcome the intensive computation of local line density and the excessive shape distortion of NLN, we tested some alternative methods based on global transformation, including a moment-based linear transformation and two nonlinear methods based on quadratic curve fitting. The alternative methods are simpler in computation and the transformed images have more natural shapes. In experiments of HCCR on large databases, the alternative methods have yielded comparable or higher accuracies to the traditional NLN.

1. Introduction

For handwritten Chinese character recognition (HCCR), a large variety of approaches have been proposed and numerous results have been reported [1]. Among them, the main advances appeared in the 1980s, when the local stroke direction feature [2, 3] and the nonlinear normalization (NLN) based on line density equalization [4, 5, 6] were proposed and got popularity. Now, these two key techniques are being used in most HCCR systems.

Compared to linear normalization, the NLN largely reduces the within-class shape variation of character images such that the recognition accuracy is improved significantly. Despite that the NLN is popularly used, it has some drawbacks as a local transformation. It may deform the character shape excessively especially for the characters of simple structures. Also, the computation of local line density is time-consuming. This paper describes our attempts to alleviate these two problems, using alternative global shape transformation methods.

For alphanumeric character recognition, some methods have been proposed to overcome the insufficiency of linear normalization [7, 8]. In HCCR, there have also been some attempts to improve the traditional NLN. The cosine function transformation of Guo et al. is a global transformation method wherein the parameters of cosine functions are determined by trials of character matching [9]. Horiuchi et al. proposed a two-dimensional NLN which complicates the shape distortion and computation of traditional NLN, though it effects in improving the recognition accuracy [10].

We intend to find alternatives to the traditional NLN which can overcome the excessive distortion of character images and the computational complexity of line density while the recognition accuracy is preserved or improved. We propose three normalization methods, including the linear moment-based method and two nonlinear methods based on quadratic curve fitting. All of them are global transformation methods, so the transformed character images look more natural than those given by local transformation. The parameters of transformation are estimated from one-dimensional moments, so the computation cost is much lower than that of local line density.

To evaluate the recognition performance of the normalization methods, we conducted experiments on two large databases of handwritten characters, ETL9B and JEITA-HP, collected in Japan. As result, the accuracy of the alternative methods are comparable to or higher than that of traditional NLN.

2. Normalization Methods

2.1. Nonlinear normalization

Normalization is to align the size, position, and shape of character images so as to reduce the within-

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)

class shape variation. In general, normalization is performed by mapping the input character image onto a standard image plane of fixed dimension. Denote the input image and the normalized image as $f(x,y)$ and $g(x',y')$, respectively, normalization is implemented by coordinate mapping

$$\begin{cases} x' &= x'(x,y), \\ y' &= y'(x,y). \end{cases}$$

The mapped coordinates (x',y') are discretized and interpolated to generate normalized image $g(x',y') = f(x,y)$.

The simple linear normalization is not sufficient to absorb the shape variation of divergent writing styles. Considering that in Chinese characters, the shape variation mainly lies in the un-uniformity of stroke distribution, Yamada et al. and Tsukumo et al. proposed the nonlinear normalization (NLN) using line density equalization [4, 5]. NLN is implemented by mapping

$$\begin{cases} x' &= W_2 \sum_{u=0}^{x} h_x(u), \\ y' &= H_2 \sum_{v=0}^{y} h_y(v), \end{cases} \tag{1}$$

where W_2 and H_2 are the width and height of the normalized image, $h_x(x)$ and $h_y(y)$ are the normalized line density histograms of x direction and y direction, respectively, which are obtained by normalizing the projections of local line densities into unity of sum.

By Tsukumo et al., the local line densities d_x and d_y are taken as the reciprocal of horizontal/vertical run-length in background area, or a small constant in stroke area [4]. While by Yamada et al., d_x and d_y are calculated by considering both background run-length and stroke run-length, and are unified to render $d_x(x,y) = d_y(x,y)$ [5]. The two methods give comparable recognition performance but the method of Tsukumo et al. is more efficient in computation [6]. For comparison, the method of Tsukumo et al. is tested in our experiments.

2.2. Alternative methods

Since line density equalization is a local transformation, the NLN may distort the character shape excessively, especially to the characters of small number of strokes. Also, the computation of local line density is time-consuming, as compared to linear normalization. The moment method and curve fitting methods described in the following are aimed to alleviate these problems.

The effect of moment normalization is to align the centroid of input image to the geometric center of normalized plane and to scale the image according to second-order one-dimensional moments. The deskewing or rotation of image involves the computation of cross two-dimensional moment, but is not necessary for Chinese characters. Formally, one-dimensional moments are computed from the projections $f_x(x)$ and $f_y(y)$ by

$$\begin{cases} \mu_{20} &= \sum_x (x-x_c)^2 f_x(x), \\ \mu_{02} &= \sum_y (y-y_c)^2 f_y(y), \end{cases}$$

where (x_c, y_c) are the coordinates of centroid.

In coordinate mapping, the centroid (x_c, y_c) is shifted to the center of normalized plane $(x'_c, y'_c) = (W_2/2, H_2/2)$. For scaling, the width and height of input image are re-determined according to the second-order moments:

$$\begin{cases} \delta_x &= \alpha\sqrt{\mu_{20}}, \\ \delta_y &= \alpha\sqrt{\mu_{02}}, \end{cases}$$

and consequently, the image boundaries are re-set to $[x_c - \delta_x/2, x_c + \delta_x/2]$ and $[y_c - \delta_y/2, y_c + \delta_y/2]$. Finally, the coordinate mapping of *Moment Normalization* is given by

$$\begin{cases} x' &= (x-x_c)\frac{W_2}{\delta_x} + x'_c, \\ y' &= (y-y_c)\frac{H_2}{\delta_y} + y'_c. \end{cases} \tag{2}$$

In normalization based on curve fitting, a quadratic function is estimated for the horizontal/vertical axis to align three anchor points (lower bound, centroid, upper bound) of input image to the three corresponding points of normalized image. In the horizontal axis, the three points of input image are $\{x_l, x_c, x_u\}$ and those of normalized image are $\{0, W_2/2, W_2\}$. The three points are aligned by fitting a quadratic function $u(x) = a_1x^2 + b_1x + c_1$. Similarly, a quadratic function $v(y) = a_2y^2 + b_2y + c_2$ is fitted for the vertical axis. Finally, the coordinate mapping is given by

$$\begin{cases} x' &= u(x), \\ y' &= v(y). \end{cases} \tag{3}$$

Fig. 1 shows some quadratic curves, in which both the input coordinate and the output coordinate have been re-scaled to $[0,1]$. We can see that when the centroid of input is centered at the midway of two bounds 0 and 1, the mapping function is linear, otherwise is a quadratic curve. When the centroid is too far from the midway, however, the output coordinate will be out of the range $[0,1]$. In this case, the curve is replaced by the closest in-range one (critical curve). For example, in Fig. 1, the critical curves (b) and (d) are used to replace out-of-range curves (a) and (e), respectively. In Chinese character images, the centroid is rarely so far

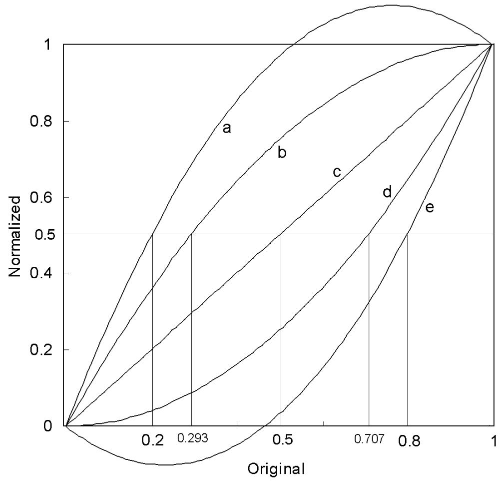

Figure 1. Quadratic curve fitting

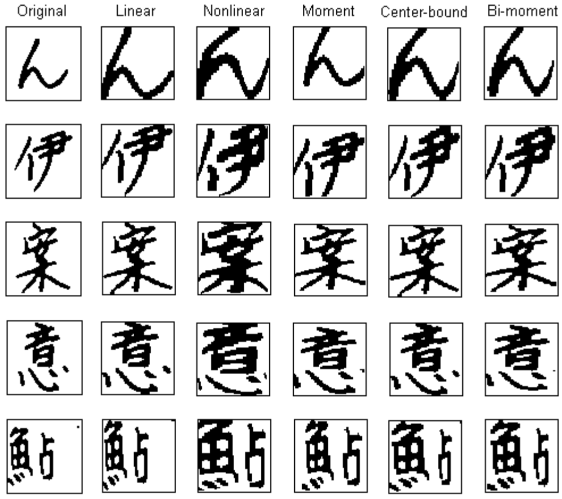

Figure 2. Examples of character shape normalization

from the midway, so the critical curves are invoked not often.

Coordinate mapping by quadratic curve fitting is applicable to the case that the centroid of input image is asymmetric with respect to the character boundaries. In the moment normalization, we have set the two bounds of each axis equally distant from the centroid. However, when we take the natural boundaries of strokes, the centroid is mostly asymmetric. This asymmetry can be corrected by quadratic curve fitting. We refer to the curve fitting method based on natural boundaries as *Centr-Bound Method*. In the following, we will introduce another method for re-setting the boundaries, which we refer to as *Bi-moment Method*.

In the bi-moment method, we compute one-sided second-order moments instead of two-sided ones:

$$\begin{cases} \mu_x^+ &= \sum_{x>x_c}(x-x_c)^2 f_x(x), \\ \mu_x^- &= \sum_{x<x_c}(x-x_c)^2 f_x(x), \\ \mu_y^+ &= \sum_{y>y_c}(y-y_c)^2 f_y(y), \\ \mu_y^- &= \sum_{y<y_c}(y-y_c)^2 f_y(y). \end{cases} \tag{4}$$

The bounds of input image are re-set to $[x_c - \delta_x^-, x_c + \delta_x^+]$ and $[y_c - \delta_y^-, y_c + \delta_y^+]$, where

$$\begin{cases} \delta_x^- = \beta\sqrt{\mu_x^-}, \\ \delta_x^+ = \beta\sqrt{\mu_x^+}, \\ \delta_y^- = \beta\sqrt{\mu_y^-}, \\ \delta_y^+ = \beta\sqrt{\mu_y^+}. \end{cases}$$

The re-set bounds and the centroid are used to estimate the quadratic functions for coordinate mapping.

In the implementation of moment and bi-moment methods, the constants α and β are set to 4 and 2, respectively.

2.3. Implementation notes

In our experiments, the normalized plane is set to 64×64 pixels. To alleviate the deformation of elongated shapes, we partially preserve the aspect ratio of input image, in so-called aspect ratio adaptive normalization (ARAN) [11]. In ARAN, the aspect ratio R_2 of normalized image is a continuous function of the aspect ratio R_1 of input image. We adopt the aspect ratio function

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

R_1 is calculated by

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

where W_1 and H_1 are the width and height of input image, respectively. When $R_2 < 1$, the normalized image is centered in the square normalized plane. For moment and bi-moment methods, W_1 and H_1 are reset to the distance between the new bounds computed from moments.

Fig. 2 show some examples of normalization by five methods: linear, NLN, moment, centr-bound, and bi-moment. We can see that the global transformation of bi-moment normalization give very similar shape correction effect to the NLN while the normalized shapes look more natural.

3. Experimental Results

To evaluate the performance of the normalization methods, we conducted experiments of HCCR on two large databases: ETL9B and JEITA-HP. The ETL9B database was collected and released by the Electro-Technical Laboratory (ETL) of Japan[1]. It contains the binary images of 3,036 characters (including 71 hiragana and 2,965 JIS level-1 Kanji), 200 images per category. We use the first 20 and 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. The writers of DATASET-B were taught to write carefully, but no constraint was imposed to the writers of DATASET-A. In principle, the dataset of each writer contains 3,306 images of 3,214 categories, in which each Kanji character was written once, while each of kana/alphanumeric was written twice.

Experimental results on JEITA-HP database have been reported by Kawatani [12]. To compare the results of JEITA-HP with those of ETL9B, we pick up the 3,036 categories of ETL9B from JEITA-HP, and for the hiragana characters, we use one image from each writer. Let us refer to the 3,036 images of one writer as a set. We use 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 dataset and test dataset are 1,541,906 and 303,334, respectively. Fig. 3 shows some images of two writers of DATASET-A and DATASET-B, respectively.

From a character image, direction features are extracted by normalization-based feature extraction (NBFE) or normalization-cooperated feature extraction (NCFE) [13]. By NBFE, the contour pixels of normalized image are assigned to four orientation planes, from each plane feature measurements are computed by blurring (Gaussian filtering and sampling) [14]. While by NCFE, the contour pixels of input image are directly assigned to standard orientation planes incorporating coordinate mapping. From each orientation plane, we extract $8 \times 8 = 64$ measurements, and in total, 256 measurements compose a feature vector. Power transformation is imposed on each measurement ($y = x^{0.5}$) to improve the Gaussianity of feature distribution [15].

[1] ETL was recently re-organized to AIST (National Institute of Advanced Industrial Science and Technology)

Figure 3. Samples of JEITA-HP database (writers A-492 and B-99)

Table 1. Recognition rates (%) of ETL9B database

NBFE	Euclid	LDF	MQDF	CPU (ms)
Linear	86.48	90.08	97.02	0.629
Nonlinear	**95.77**	96.34	98.74	0.768
Moment	95.22	96.60	98.71	0.663
Centr-Bound	94.12	95.96	98.48	0.671
Bi-moment	95.49	**96.65**	**98.77**	0.667
NCFE	Euclid	LDF	MQDF	CPU (ms)
Linear	86.92	90.28	97.02	0.169
Nonlinear	**96.55**	96.76	**98.84**	0.338
Moment	95.71	96.80	98.73	0.201
Centr-Bound	94.66	96.16	98.54	0.220
Bi-moment	95.96	**96.85**	98.80	0.210

We give the results of three classifiers, namely, Euclidean distance to mean, linear discriminant function (LDF), and modified quadratic discriminant function (MQDF2) [3]. The MQDF2 was proposed by Kimura et al. to reduce the storage and computation of ordinary QDF and to improve the classification performance. In MQDF2, the covariance matrix of each class is regularized by replacing the minor eignvalues with a constant. In our experiment, the number of principal eigenvectors is set to 50. Since the computation of MQDF2 is expensive for large category recognition, a number of candidate classes are selected by LDF and then MQDF2 is computed only on the candidate classes. The number of candidates is set to 100 for linear normalization and 50 for other methods.

The recognition rates of of two databases are shown

Table 2. Recognition rates (%) of JEITA-HP database

NBFE	Euclid	LDF	MQDF	CPU (ms)
Linear	79.57	84.93	95.74	0.643
Nonlinear	**91.00**	92.83	97.44	0.741
Moment	90.59	93.39	97.63	0.676
Centr-Bound	88.92	92.37	97.39	0.663
Bi-moment	90.98	**93.48**	**97.67**	0.625
NCFE	Euclid	LDF	MQDF	CPU (ms)
Linear	80.83	85.43	95.92	0.102
Nonlinear	**92.50**	93.62	97.76	0.238
Moment	91.62	93.76	97.75	0.152
Centr-Bound	90.10	92.79	97.52	0.146
Bi-moment	92.04	**93.86**	**97.81**	0.132

in Table 1 and Table 2, respectively. The average CPU time (on Pentium-4-1.90GHz) of normalization is also given to compare the computational complexity. From the results of either ETL9B or JEITA-HP, we can see that the NLN improves the recognition accuracy significantly compared to linear normalization. The recognition rate of bi-moment method is comparable to or higher than that of NLN. The accuracy of moment normalization is higher than that of centr-bound method, and both are considerably higher than that of linear normalization. In comparison of NLN and bi-moment normalization, the NLN wins with Euclidean distance classifier while in other cases, bi-moment normalization wins over NLN except once.

In respect of processing time, it is evident that NCFE saves much time compared to NBFE. For either NBFE or NCFE, the difference of CPU time between NLN and other methods is around 0.1ms. This difference mainly lies in the calculation of coordinate mapping. Note the NLN was tested under a fast implementation of the method of Tsukumo et al. [4], which is much faster than that of Yamada et al. [5].

4. Conclusion

This paper described our attempts of finding substitutes for nonlinear normalization in HCCR. The alternative methods (moment normalization, centr-bound, and bi-moment method) generate more natural normalized shapes and are less complicated than the NLN. In experiments, the recognition accuracy of bi-moment method is comparable or higher than that of NLN, and the accuracy of moment normalization (which is actually linear) is competitive.

References

[1] T.H. Hilderbrand, W. Liu, Optical recognition of Chinese characters: advances since 1980, *Pattern Recognition*, 26(2): 205-225, 1993.

[2] M. Yasuda, H. Fujisawa, An improvement of correlation method for character recognition, *Trans. IEICE Japan*, J62-D(3): 217-224, 1979.

[3] F. Kimura, K. Takashina, S. Tsuruoka, Y. Miyake, Modified quadratic discriminant functions and the application to Chinese character recognition, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 9(1): 149-153, 1987.

[4] J. Tsukumo, H. Tanaka, Classification of handprinted Chinese characters using non-linear normalization and correlation methods, *Proc. 9th ICPR*, Rome, Italy, 1988, pp.168-171.

[5] 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.

[6] S.-W. Lee, J.-S. Park, Nonlinear shape normalization methods for the recognition of large-set handwritten characters, *Pattern Recognition*, 27(7): 895-902, 1994.

[7] G. Nagy, N. Tuong, Normalization techniques for handprinted numerals, *Communi. ACM*, 13(8): 475-481, 1970.

[8] R.G. Casey, Moment normalization of handprinted character, *IBM J. Res. Dev.*, 14: 548-557, 1970.

[9] J. Guo, et al., Recognition of handwritten characters using pattern transformation method with cosine function, *Trans. IEICE Japan*, J76-D-II(4): 835-842, 1993.

[10] T. Horiuchi, R. Haruki, H. Yamada, K. Yamamoto, Two-dimensional extension of nonlinear normalization method using line density for character recognition, *Proc. 4th ICDAR*, Ulm, Germany, 1997, pp.511-514.

[11] C.-L. Liu, M. Koga, H. Sako, H. Fujisawa, Aspect ratio adaptive normalization for handwritten character recognition, *Advances in Multimodal Interfaces—ICMI2000*, LNCS 1948, T. Tan, Y. Shi, and W. Gao (Eds), Springer, 2000, pp.418-425.

[12] T. Kawatani, H. Shimizu, Handwritten Kanji recognition with the LDA method, *Proc. 14th ICPR*, Brisbane, 1998, Vol.II, pp.1031-1035.

[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, 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.

[15] T. Wakabayashi, S. Tsuruoka, F. Kimura, Y. Miyake, On the size and variable transformation of feature vector for handwritten character recognition, *Trans. IEICE Japan*, J76-D-II(12): 2495-2503, 1993.