3

Adaptive Feature Extraction Method for Degraded Character Recognition

Minoru Mori, Minako Sawaki and Junji Yamato
NTT Communication Science Laboratories (NTT Corporation)
Japan

1. Introduction

Most character recognition applications target machine printed and handwritten characters on paper documents. Recently, the recognition of text in videos, web documents, and natural scenes has become an urgent demand; research has intensified because this task is difficult to realize (Antonacopoulos & Hu, 2004; Doermann et al., 2003; Kise & Doermann, 2007; Lienhart & Wernicke, 2002; Lyu et al., 2005; Zhang & Kasturi, 2008). The problems posed by recognizing low quality characters in the above mentioned applications are mainly due to deformation such as the variety of font styles and style effects, as well as image degradation like background noise, blur, and low resolution. A key weakness of most conventional character recognition methods is that they tackle either one problem or the other, not both.

For overcoming image degradation, some methods, e.g. (Ho, 1998; Kopec, 1997; Xu & Nagy, 1999), design templates that reflect the degradation type anticipated. Also a robust discriminant function for recognizing degraded characters was proposed in (Sato, 2000; Sawaki & Hagita, 1998). Unfortunately, these methods are sensitive to shape deformation, since they employ image-based template matching. They fail to effectively handle multiple fonts and several style effects.

On the other hand, geometric features are often used for recognizing multiple fonts. Stroke direction is particularly effective against character deformation (Umeda, 1996). For example, the direction contribution based on stroke run-length is effective (Akiyama & Hagita, 1990; Srihari et al., 1997; Zhu et al., 1997). However, geometric features are not robust against corruption of information due to image degradation. In addition, although geometric features are more robust against deformation than image-based template matching, they are not invariant for deformation such as aspect ratio fluctuation and stroke position shift. Therefore, geometric features are weak against the kinds of deformation that are not present in the training samples.

For overcoming deformation problems mentioned above, nonlinear shape normalized techniques (Tsukumo & Tanaka, 1988; Yamada et al., 1990) have been proposed as a pre-processing method to relocate strokes uniformly. They normalize a pattern by exploiting the distance between strokes (Tsukumo & Tanaka, 1988) and stroke line density (Yamada et al., 1990), and are mainly aimed at the recognition of Kanji characters that consist of many strokes in mostly square patterns. Therefore, applying these methods to the recognition of numerals, alphabets and kana characters, which consist of fewer strokes and are not square shape, is difficult. Also these methods are ineffective for degraded characters with backgrounds noise and blur because when calculating stroke line density or distance between strokes they basically assume that characters are not degraded.

To reduce the influence caused by image degradation in the recognition based on geometric features, some methods try to compensate for the inaccuracy in the values of discriminant function or geometric feature by assuming the type of degradation and estimating the degree of degradation using local pixel distributions (Mori et al., 2001; Omachi et al., 2000). The method in (Omachi et al., 2000) detects blurred areas using the thinning technique and compensates similarity values in those areas. Another approach (Mori et al., 2001) offsets the feature values using the complexity of pixel distributions. They, however, are counterproductive when the assumption of the degradation type is invalid. This suggests the difficulty of compensating geometric information using local pixel distributions; discriminating noise from the strokes of the character is almost impossible.

To tackle the problems mentioned above, we focus on a category-dependent method with the top-down approach. All category-dependent methods assume the category of an input pattern and adaptively compensates deformation or image degradation by exploiting category-specific information. Category-dependent methods include a shape normalization method that tackles the deformation (Nakagawa et al., 1999; Wakahara & Odaka, 1998). In this chapter we propose a category-dependent method that achieves robustness against both deformation and image degradation (Mori et al., 2005; 2010). Our method estimates the degree of deformation and degradation of the input pattern on the basis of specific information of each category. Exploiting the category information enables us to extract the variation of the aspect ratio and that of the run-length used for computing feature values. The fluctuations in shape and feature values are then offset by the estimated compensation coefficients. To evaluate the proposed method, we apply it to the recognition of video text that is degraded by background noise and blur, and deformed by aspect ratio fluctuations.

The rest of the chapter is organized as follows: Section 2 provides a description of the directional feature and the algorithm of the proposed method. Experimental results gained from video text are reported in Section 3. Section 4 summarizes this chapter.

2. Adaptive Feature Extraction Using Category Information

2.1 Overview

This section describes the geometric feature used and details the algorithm of our method; feature extraction that exploits category-specific information. We use the stroke directional information based on the stroke run-length as the geometric feature. The proposed method tackles deformation and image degradation in two ways. One is adaptive normalization. Adaptive normalization is applied after the classification stage, and yields a normalization size appropriate for the input pattern with fluctuation in aspect ratio by repeating the processes of normalization and classification. The other is feature compensation. Feature compensation is applied to the candidates output by the classification stage, and offsets the feature values corrupted by image degradation to obtain higher recognition accuracy in the final recognition stage. Figure 1 overviews the process flow including the proposed method.

2.2 Directional Feature

Geometric features that extract stroke direction are effective for discriminating multiple fonts. In this chapter we use the stroke directional feature (Akiyama & Hagita, 1990; Srihari et al., 1997; Zhu et al., 1997) that is based on stroke run-length. This feature is extracted as follows:

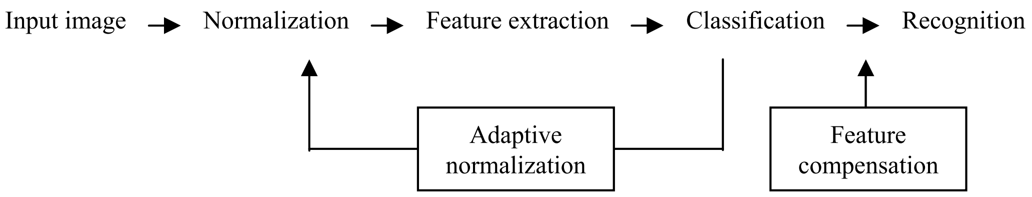


Fig. 1. Overview of processing.

Let l_1, l_2, l_3, and l_4 be the run-lengths on the horizontal, right diagonal, vertical, and left diagonal directions at each black pixel of strokes, respectively. Let $l_{m,i}$ be the run-length yielded by averaging l_i on the m-th block obtained by partitioning a pattern. Let $d_{m,i}$ be the degree of contribution in stroke direction as components of feature vector for the m-th block. $d_{m,i}$ can be computed by the following steps.

Step 1: The input pattern is divided into $N \times N$ blocks.

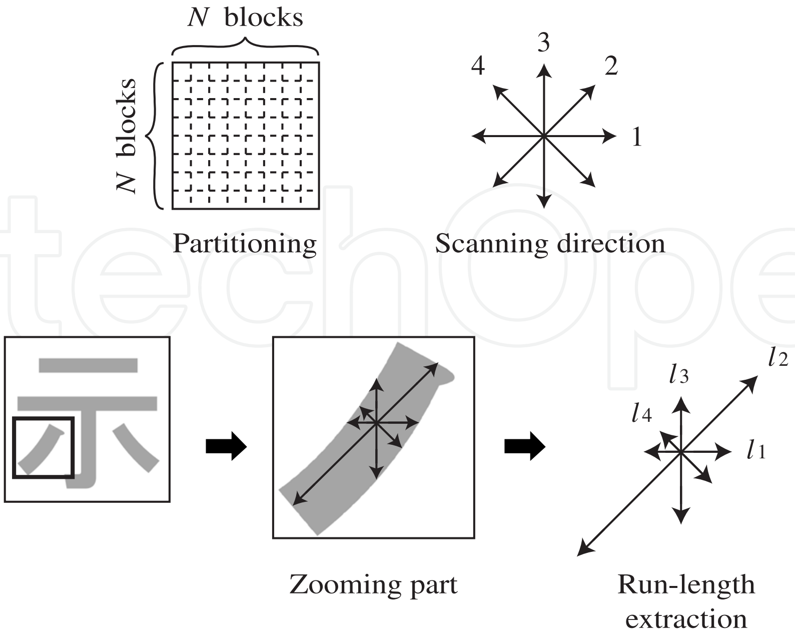
Step 2: l_i $(i = 1, ..., 4)$ is extracted at each black pixel.

Step 3: $l_{m,i}$ $(m = 1, ..., N \times N)$ is calculated by averaging l_i on each block.

Step 4: $d_{m,i}$ is computed on each block by

$$d_{m,i} = \frac{l_{m,i}}{\sqrt{\sum_{j=1}^{4} {l_{m,j}}^2}}. \tag{1}$$

Here we use $N = 8$. Figure 2 shows each step in the extraction of stroke directional feature from an input pattern.



Partitioning

Scanning direction

Zooming part

Run-length extraction

Fig. 2. Directional feature extraction.

2.3 Adaptive Normalization

Characters used in videos or natural scenes come in various fonts and are often deformed when they are superimposed or aligned. The fluctuation in aspect ratio based on these diversities is one of the factors that degrade the recognition accuracy. However Japanese characters contain so many various structures and ratios, that estimating the most appropriate ratio is difficult. The shape normalization to a pre-defined aspect ratio often normalizes a pattern such that it approaches an erroneous category which degrades the recognition accuracy. To normalize a pattern effectively, the shape normalization methods proposed in (Nakagawa et al., 1999; Wakahara & Odaka, 1998) use templates for each category and are effective for normalizing such deformed characters. These methods, however, are too time-consuming because they produce normalized patterns for every each category. Here we propose an adaptive normalization scheme that uses category-specific information; it is simple but effective for compensating aspect ratio fluctuation (Mori et al., 2010). Figure 3 shows the flow of the adaptive normalization scheme.

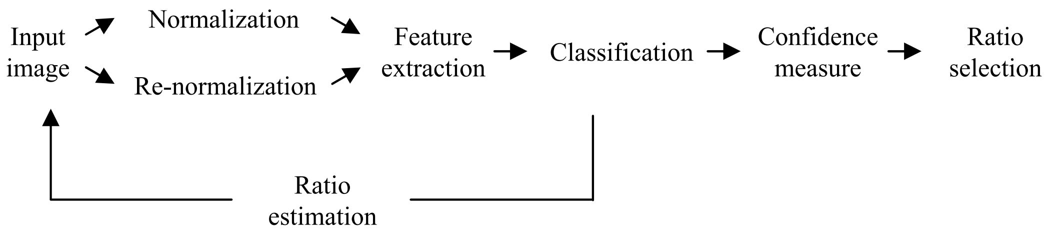


Fig. 3. Flow of adaptive normalization.

Our proposal uses the ratio information of training samples of each category and is applied after the first classification stage as follows: First the input pattern is normalized for the predefined size retaining the aspect ratio of the input pattern; Here let r_x^0 and r_y^0 be the horizontal and vertical rectangular size of the input pattern, respectively. Let R be the pre-defined size for pattern normalization, and $\max(,)$ be the operation that returns the larger element. The horizontal and vertical rectangular dimensions of the normalized pattern, r_x and r_y, are given by

$$r_x = r_x^0 \cdot R/\max(r_x^0,\ r_y^0), \tag{2}$$

$$r_y = r_y^0 \cdot R/\max(r_x^0,\ r_y^0). \tag{3}$$

When $R < \max(r_x^0, r_y^0)$, the input pattern is scaled down so that the longer rectangular size fits pre-defined size R, and otherwise the input pattern is scaled up. Next the normalized pattern is classified and candidate categories are obtained. Let r_x^c and r_y^c be the rectangular sizes for the c-th category obtained by averaging r_x and r_y of training samples in the c-th category, where $c\ (= 1, ..., C)$ denotes the category number. Here we define the new rectangular sizes, r'_x and r'_y, by averaging r_x^c and r_y^c among the top candidates as follows:

$$r'_x = \frac{1}{N1} \sum_{c=1}^{N1} r_x^c, \tag{4}$$

$$r'_y = \frac{1}{N1} \sum_{c=1}^{N1} r_y^c, \tag{5}$$

where $N1$ is the number of the candidate categories used for calculating new rectangular sizes. Finally, the input pattern is re-normalized to fit the size of r'_x and r'_y and re-classified; When $r'_x > r^0_x$, the horizontal rectangular size of the input pattern is enlarged by the factor of r'_x/r^0_x, otherwise shrunk by r'_x/r^0_x times. The new vertical rectangular size is obtained in the same manner. The new candidate categories are obtained by re-classifying the re-normalized pattern.

It should be noted here that when the classification result involves many error candidates, the normalization of input pattern tends to result in an erroneous size or shape. To avoid overfitting to erroneous sizes and obtain appropriate values, we define the confidence measure, s_{conf}, as follows:

$$s_{conf} = \sum_{c=1}^{N2} dist_1/dist_c, \tag{6}$$

where $dist_c$ is the distances obtained in the classification stage for the c-th candidate category. $N2$ is the number of categories used for calculating the confidence measure. s_{conf} is defined as the summation of the ratio between the 1st candidate's distance and the c-th candidate's one and means the reliability of the classification result. We can select the appropriate normalization pattern using this measure. When s_{conf} obtained using the first normalized pattern is less than that of the re-normalized one, the ratio of the first normalized one is more reliable and so the first normalized pattern is selected as indicating the appropriate rectangular size. Otherwise, the re-normalized pattern is selected. The normalized pattern with the selected aspect ratio is submitted for the following stage.

2.4 Feature Compensation

Feature values extracted from a degraded pattern are often corrupted and cause mis-recognition. To tackle this problem, we introduce a feature compensation technique that estimates the degree of degradation in the input pattern (Mori et al., 2005; 2010). Figure 4 shows the flow of feature compensation technique.

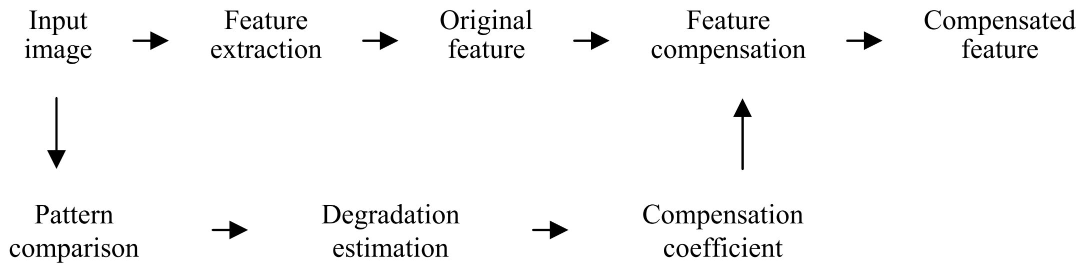


Fig. 4. Flow of feature compensation.

Feature values extracted from parts degraded like background noise or blur and those extracted from strokes are generally combined. In other words, the influence of degradation appears as a weight that depends on the degree of degradation. Therefore, by estimating the degree of degradation, we can acquire the compensation coefficient needed to compensate the degraded feature values. This estimation thus enables us to obtain the most approximate feature vector by compensating the degraded feature vector.

The key to estimating the degree of degradation is using the variation in run-length distribution of the input pattern. The variation of run-length basically depends on the degree of degradation. Therefore, the degree of degradation can be estimated by extracting the degree of variation in run-length distribution of the input pattern. However, as mentioned in Section 1, it is impractical to estimate the variation from just pixel distribution in the input pattern. To realize this estimation, we exploit the template of each category as category-specific information. Comparing the input pattern to the template of each category enables us to calculate the variation of run-length against the focused category.

It should be noted that using a run-length distribution extracted from just one localized region tends to result in failure. The reason is that local parts of different categories often have similar properties, particularly when the input pattern is degraded. Therefore, we estimate the degree of degradation from the total pattern, not just the local regions. Adjusting the local estimation against the global estimation provides an appropriate compensation coefficient for the focused part. In summary, combining local and global estimations, both based on pattern comparison, enables us to extract an approximate feature vector even from strongly degraded characters by compensating the fluctuation in feature values.

As the template of each category, we use the directional stroke run-length. The templates for each category are obtained as follows: The averaged stroke run-length $l_{m,i}$ is calculated using same steps given in Section 2.2. The run-length vectors used as the template for the c-th category, $\bar{l}^c_{m,i}$, are then obtained by averaging $l_{m,i}$ from training samples of the c-th category.

Next we define the degree of degradation as the average of the degree of degradation from blocks obtained by partitioning the input pattern. The degree of degradation on focused block, $p^c_{m,i}$, is calculated as the ratio between the run-length distribution of the input pattern, $l_{m,i}$, and that of the c-th category's template, $\bar{l}^c_{m,i}$, as follows:

$$p^c_{m,i} = \begin{cases} (l_{m,i} - \bar{l}^c_{m,i})/l_{m,i} & \text{if } (l_{m,i} > \bar{l}^c_{m,i}) \\ (\bar{l}^c_{m,i} - l_{m,i})/\bar{l}^c_{m,i} & \text{otherwise.} \end{cases} \tag{7}$$

$p^c_{m,i}$ approaches 1 as the focused block of the input pattern become more degraded or more dissimilar. $p^c_{m,i}$ becomes 0 for the comparison of identical patterns. Also, the degree of degradation over the pattern against the c-th category, g^c, is defined by

$$g^c = \frac{\sum_{m=1}^{N^2} \sum_{i=1}^{4} p^c_{m,i}}{4 \cdot N^2}. \tag{8}$$

g^c approaches 1 as the input pattern become degraded or dissimilar. g^c becomes 0 if we are comparing the identical patterns.

The compensation coefficient is then calculated. First the compensated run-length, $l'_{m,i}$, an indication of the compensation amount, is computed using the above degree of degradation by

$$l'_{m,i} = l_{m,i} \cdot (1 - g^c) + \bar{l}^c_{m,i} \cdot g^c. \tag{9}$$

The compensation coefficient $w^c_{m,i}$ is computed from $l'_{m,i}$ in each block by

$$w^c_{m,i} = (l_{m,i} - l'_{m,i})/(l_{m,i} - \bar{l}^c_{m,i}). \tag{10}$$

Finally, a new feature value against the c-th category, $d^c_{m,i}$, is obtained by compensating $d_{m,i}$ with coefficient $w^c_{m,i}$ as follows:

$$d^c_{m,i} = d_{m,i} \cdot (1 - w^c_{m,i}) + \bar{d}^c_{m,i} \cdot w^c_{m,i} \tag{11}$$

where $\bar{d}^{c}_{m,i}$ is the mean vector of the c-th category. C feature vectors are obtained by repeating the above procedure for every category and the input pattern is recognized by calculating distances between the vector from the input pattern and the reference vector of each category. Figure 5 shows the flow of the feature extraction and the recognition stage in the proposed method and the conventional one.

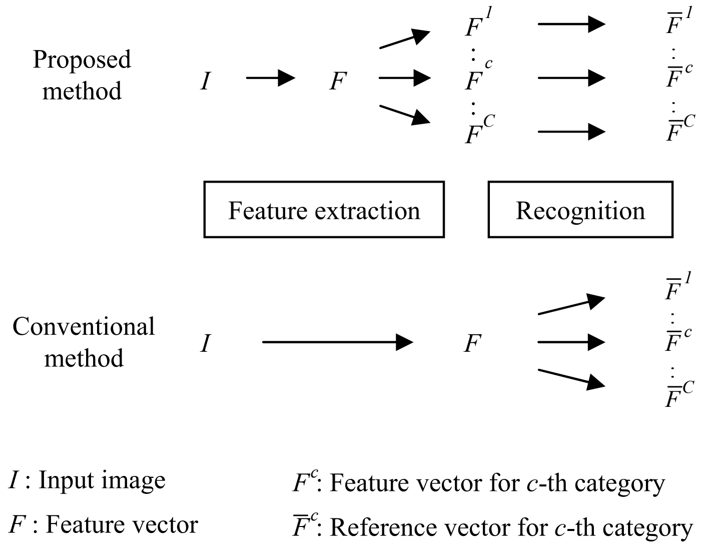


Fig. 5. Flow in feature extraction and recognition.

Figure 6 visualizes the feature values obtained using compensation technique and those of the original feature for the character with background noise. Darker block represents higher contribution strength in each stroke direction. Figure 6 shows that the compensated feature yielded by the proposed method still retains stroke direction while suppressing the influence of background noise.

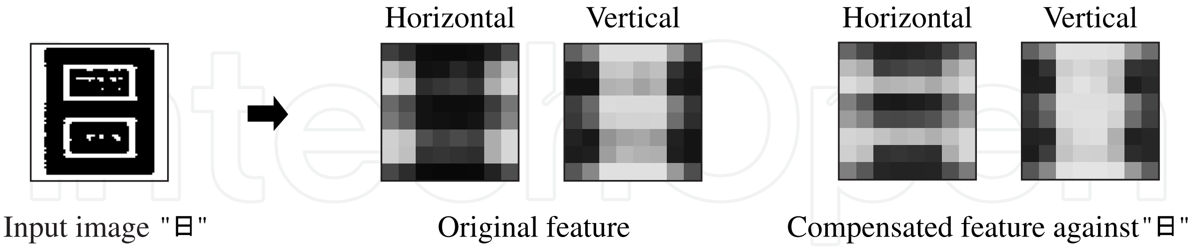
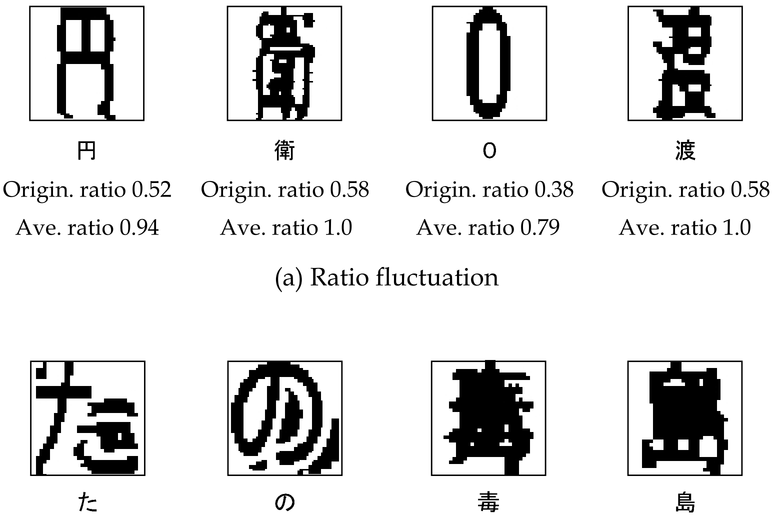


Fig. 6. Examples of feature values.

3. Recognition Experiments

3.1 Data

To confirm the proposed method's robustness against degradation and deformation, we used the characters in videos as the experimental data. Characters extracted from binarized video frames suffer from several types of degradation and deformation; Fluctuation in aspect ratio, background noise, and blur are the main causes of poor recognition accuracy. Ratio fluctuation comes from the variety of fonts used and shape adjustment caused by aligning characters in fixed space when superimposing them. Background noise is caused by misjudging the background region as character region due to similar properties such as color or size. Blur is derived from the low spatial resolution of the image and inappropriate thresholds used in binarizing the video frame. Figure 7 shows typical characters extracted from binarized frames using the method proposed in (Kuwano et al., 1997). Characters with varied aspect ratio are shown on the upper row (a) and characters with background noise or blur are shown on the lower row (b). On the upper low, each value of "Origin. ratio" indicates the original aspect ratio (horizontal size/vertical size) of the each sample, and each value of "Ave. ratio" indicates the averaged aspect ratio of training data mentioned below in each sample's category. These values show that the aspect ratio of these samples are strongly fluctuated.



(a) Ratio fluctuation

(b) Background noise / blur

Fig. 7. Characters extracted from video.

We used the following data in the recognition experiments. As the training data set, we used 67 fonts of machine-printed Japanese characters from 3,190 categories. As the test data set, 9,980 samples were selected from samples we gathered; They contained 7,841 clean / ratio-fluctuated / slightly noisy characters and 2,139 noisy / blurred ones.

3.2 Experimental Conditions

Normalization size for each sample was $R = 64$ pixels. Each feature vector consisted of 256 dimensional components (8×8 blocks $\times$ 4 directions). The dictionary was constructed by averaging features from the training samples for each category. The following Euclidean distance was used as the classifier

$$dist^c = \sqrt{\textstyle\sum_{m=1}^{64} \sum_{i=1}^{4} (d^c_{m,i} - \bar{d}^c_{m,i})^2}. \tag{12}$$

In the adaptive normalization process, the adaptation iteration was set to the 1 time. $N1$ and $N2$ were decided through a preliminary experiment. We used $N1 = 128$ for the aspect ratio estimation and $N2 = 16$ for the recognition confidence measure.

3.3 Experimental Results

First, we compared the adaptive normalization technique to conventional fixed normalization; the input pattern was normalized using a pre-defined aspect ratio. In this chapter we applied the following two normalization flows as conventional techniques; In the first one (Fixed normalization 1), multiply the shorter rectangular length by the normalization parameter, $rt(= 1.0 \sim 1.6)$, so that the input pattern becomes more square. For example, when $r_x^0 < r_y^0$, r_x and r_y are given by

$$r_x = r_x^0 \cdot R/\min(r_x^0 \cdot rt,\ r_y^0), \tag{13}$$

$$r_y = R, \tag{14}$$

where $\min(,)$ is the operation that returns the smaller element. When $r_x^0 > r_y^0$, the operation is applied to r_y^0 in the same manner. Normalization with $rt = 1.0$ yields the normalized pattern retaining the original aspect ratio of the input pattern as the standard normalization method. On the other hand, the second conventional method (Fixed normalization 2) normalizes the input pattern to a square shape; the horizontal and vertical lengths of the normalized pattern are R and the aspect ratio is constant at 1.0.

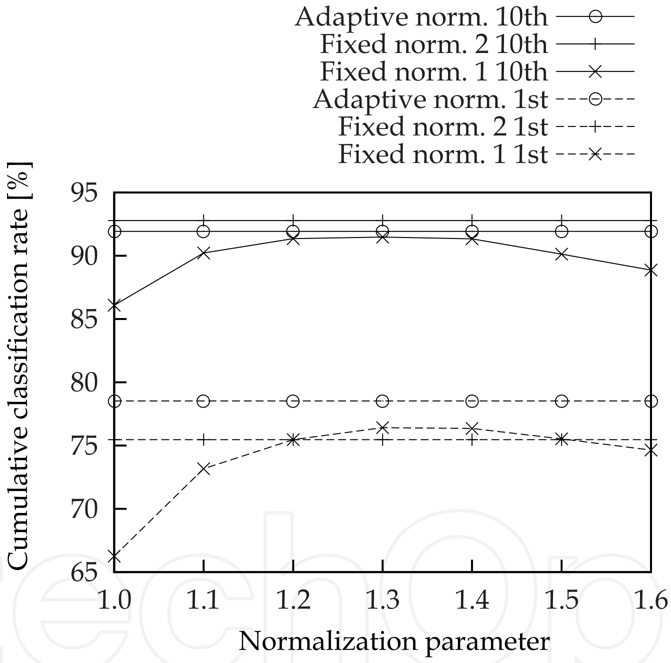


Fig. 8. Classification rates versus normalization parameter

Figure 8 shows the 1st and 10th classification rates of the adaptive normalization method and the two conventional techniques for all test data. The horizontal axis shows the normalization parameter rt for fixed normalization 1. Figure 8 indicates that the adaptive normalization yielded 12.3% better rates for the 1st classification rates and 5.8% better rates for the 10th rates than the standard normalization, $rt = 1.0$. Also the best result of the 1st and 10th classification rates obtained by $rt = 1.3$ in fixed normalization 1 and the 1st classification rate of fixed normalization 2 are lower than that offered by the adaptive normalization. These results show that the proposed adaptive normalization accurately estimates and determines the rectangular sizes for each input pattern in the presence of aspect ratio fluctuation. On the other hand, the 10th classification rate obtained by the fixed normalization 2 is only slightly higher than that of the proposed adaptive normalization method. From these results, square normalization by fixed normalization 2 seems to offset some degree of the ratio fluctuation in the compulsory normalization method. However, this method deforms patterns of different categories to a similar shape and so degrades the 1st classification rate.

Next, we examined the effectiveness of the proposed feature compensation technique. The conventional method consists of the original directional feature without compensation technique. Adaptive normalization was applied to both features. Figure 9 shows the classification rates of these methods for a data set containing only background noise and blur. The compensated feature achieved about 8% better classification rates than the original one for all candidate orders. In particular, for the top ten candidates, the compensated feature obtained 7.7% higher rates than the original one; that means our proposed method yielded 28% fewer errors than the original one. This result proves that the proposed feature compensation effectively achieves robustness against image degradation such as that caused by background noise and blur.

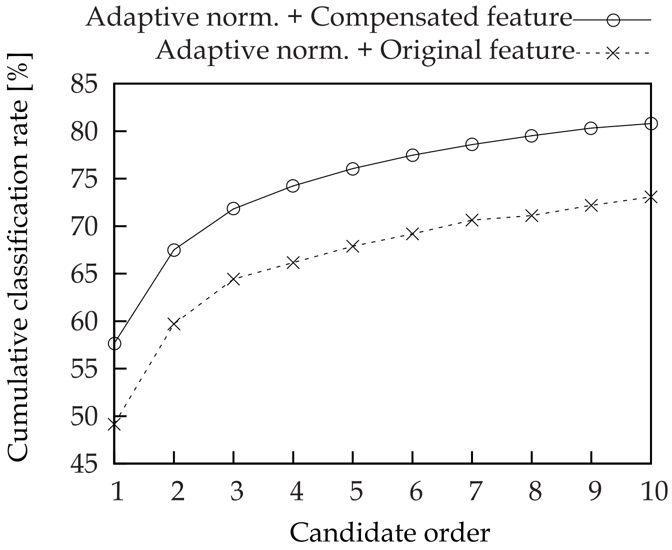


Fig. 9. Classification rates for each feature.

Finally, we evaluated the overall performance of the proposed method using all test data including clean and degraded data. Figure 10 shows the classification rates for each method. We used $rt = 1.0$ in fixed normalization 1 as the standard method retaining the original aspect ratio of the input pattern. Figure 10 shows that adaptive normalization offers significantly higher rates for every candidate order than the normalization method that holds the original aspect ratio of the input pattern. Moreover, the compensated feature yields about 2% higher classification rates than the original one for both fixed and adaptive normalization. This advantage proves that our method can effectively offset the variation in features caused by degradation without lowering the recognition accuracy for clean data. The results shown in Figure 10 mean that our proposed method is effective for both fluctuation of aspect ratio as deformation and background noise and blur as image degradation.

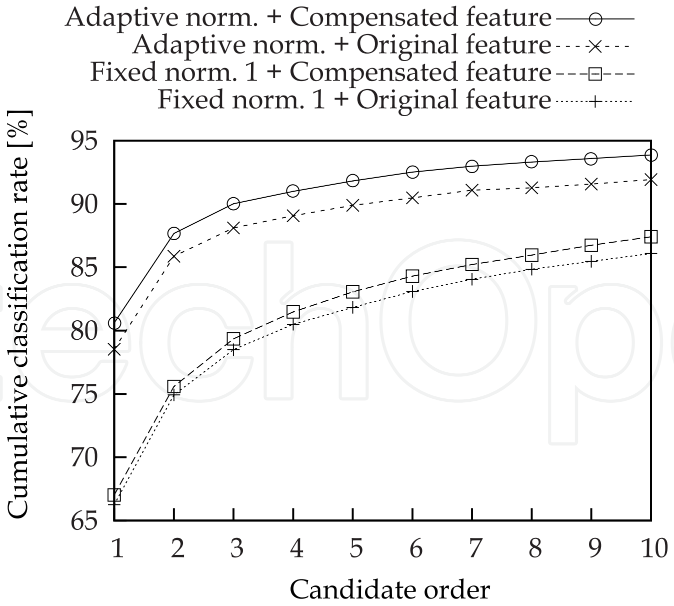


Fig. 10. Classification rates for all test data.

3.4 Discussion

We first evaluated the effect of ratio selection using the confidence measure. Table 1 shows the classification rates with/without the ratio selection. From Table 1, the use of ratio selection raised the recognition accuracy for both 1st and 10th rates. This shows that the proposed confidence measure and ratio selection procedure are effective for avoiding over-fitting to erroneous sizes.

	1st rate	10th rate
With ratio selection	78.53%	91.93%
Without ratio selection	75.20%	90.56%

Table 1. Classification rates with/without ratio selection.

Next, we compared the classification rates obtained by adaptive normalization to those obtained by fixed normalization 1 using ratio selection between $rt = 1.0$ and $rt = 1.3$ to examine the effectiveness of aspect ratio estimation. Table 2 shows the 1st and 10th classification rates for each method. From Table 2, the aspect ratio estimation yielded more appropriate ratios automatically and so raised the recognition accuracy. It should be noted that it's difficult to know parameter $rt = 1.3$ for the best rates in fixed normalization 1 in advance.

	1st rate	10th rate
Adaptive norm.	78.53%	91.93%
Fixed norm. 1 with ratio selection	78.12%	91.34%

Table 2. Classification rates using adaptive normalization and fixed normalization 1 with ratio selection.

Then, we examined the classification rates in repeating the adaptive normalization for validating the effect of the normalization iteration. Figure 11 shows the 1st and 10th classification rates for each iteration with the adaptive normalization. When the iteration time, $N1$, is more than 1, the classification rates for both 1st and 10th are saturated. This result indicates that the adaptive normalization effectively estimates the rectangular sizes but has also limited ratio estimation ability.

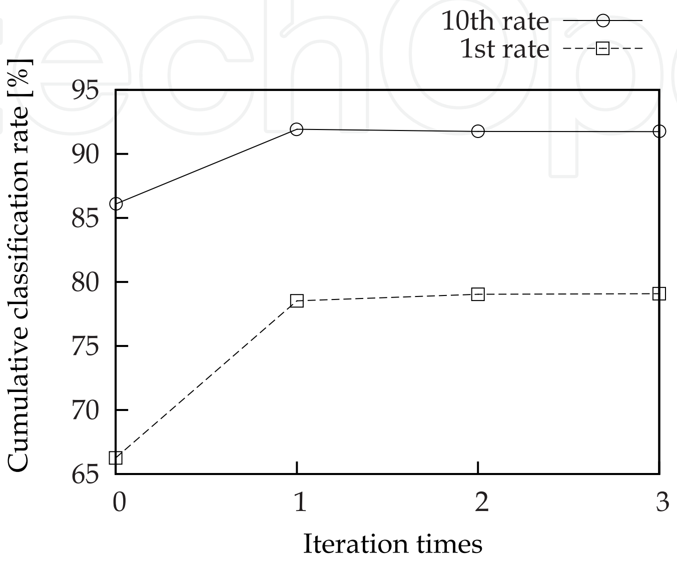


Fig. 11. Classification rates for each iteration time.

Moreover, we compared the CPU run time required for the recognition with adaptive normalization to that with fixed normalization. The system resources and development environment are as follows:

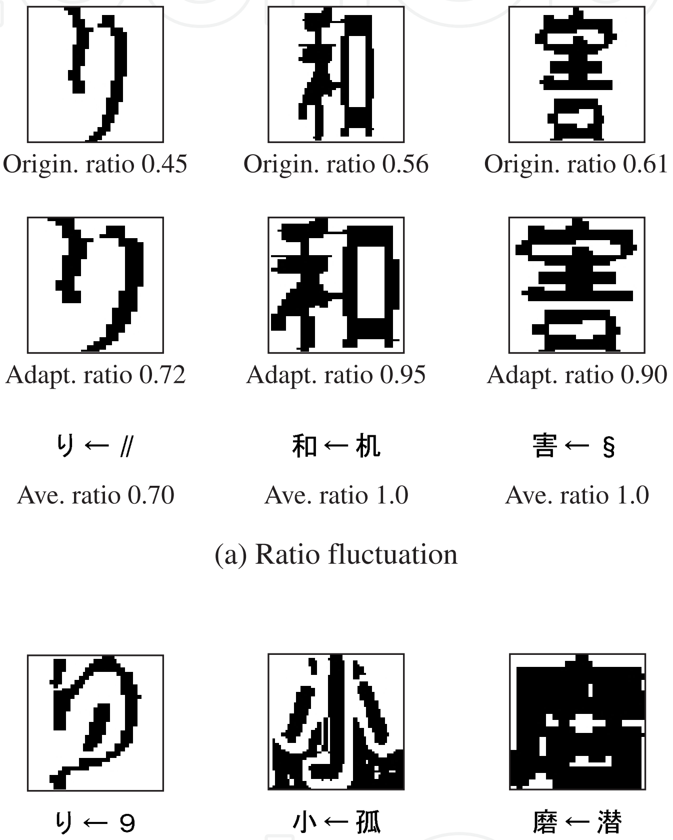
- CPU: Core2 Duo E6600 2.4GHz
- Memory: 1.5GB
- OS: Windows XP
- Language: C/C++

Table 3 shows the CPU run time per sample for each normalization method. The process assessed ran from pattern normalization to classification, and the CPU run time was computed by averaging the time taken to process each sample in the complete test data set. Table 3 shows that adaptive normalization has consumes more CPU run time. The increase is caused by the repetition of feature extraction and classification and the addition of the ratio selection process. However, this increase in time is offset by the increase in recognition accuracy.

	CPU run time
Adaptive normalization	3.13 msec
Fixed normalization	1.97 msec

Table 3. CPU run time with each normalization method.

Figure 12 shows examples recognized correctly by the proposed method which were recognized erroneously by the conventional one (correct result ← erroneous result). Figure 12 (a) shows examples with ratio fluctuation and Figure 12 (b) shows examples with background noise or blur. The upper row in (a) expresses first normalized patterns and their aspect ratios. The lower one in (a) expresses adaptive normalized patterns and their aspect ratios. "Ave. ratio" means the averaged aspect ratio of training data. Those examples show that the proposed normalization method well handles aspect ratio fluctuation and can estimate the most appropriate aspect ratios. With regard to the examples in (b), the proposed method effectively compensated the feature fluctuation, and so suppressed errors.



(a) Ratio fluctuation

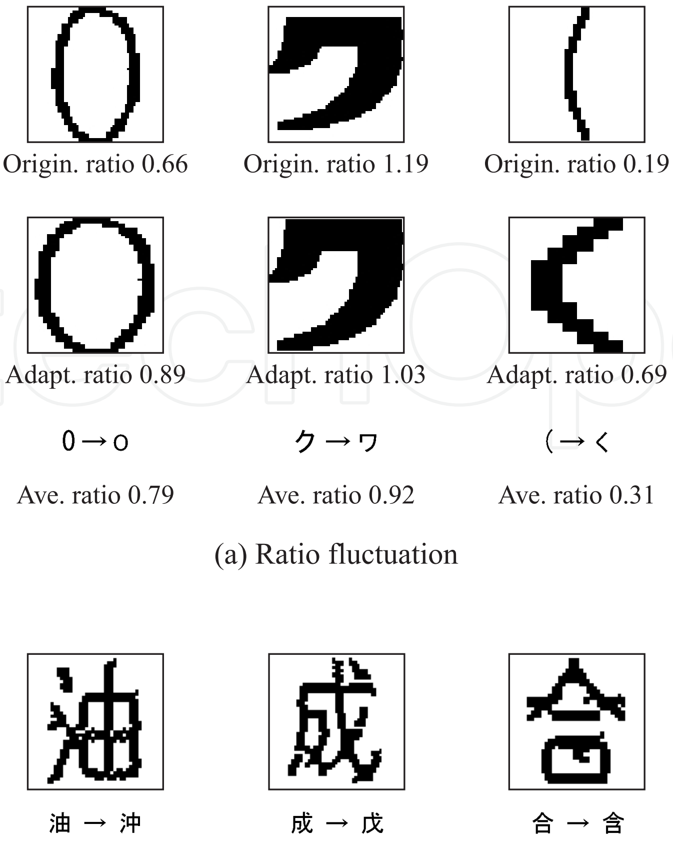
(b) Background noise / blur

Fig. 12. Examples of correct recognition.

Figure 13 shows examples recognized correctly by the original feature that were recognized erroneously by the compensated one for the first candidate (correct category → erroneous result). The errors in (a) are caused by the mis-normalization of the aspect ratio of the input pattern, it approaches erroneous category's ratio, and the failure of ratio selection using the confidence measure. The errors in (b) are caused by the compensation of feature values on the blocks deemed to be strongly degraded.

4. Conclusion

We have proposed a feature extraction method that is based on category-dependent processing for the recognition of characters exhibiting both deformation and degradation. Our



(a) Ratio fluctuation

(b) Background noise / blur

Fig. 13. Examples of mis-recognition.

method estimates the degrees of deformation and degradation of the input pattern by exploiting category-specific information. The estimation realizes adaptive compensation of aspect ratio fluctuations and feature value corruption caused by image degradation. Recognition experiments with video texts exhibiting varying levels of deformation and degradation showed that our method achieves higher classification rates than the conventional method.

5. References

Akiyama, T. & Hagita, N. (1990). Automated entry system for printed documents, *Pattern Recognition* **23**(11): 1141–1154.

Antonacopoulos, A. & Hu, J. (eds) (2004). *Web Document Analysis: Challenges and Opportunities*, World Scientific Press.

Doermann, D., Liang, J. & Li, H. (2003). Progress in camera-based document image analysis, *Proceedings of 7th International Conference on Document Analysis and Recognition*, Vol. 1, pp. 606–616.

Ho, T. (1998). Bootstrapping text recognition from stop words, *Proceedings of 14th International Conference on Pattern Recognition*, Vol. 1, pp. 605–609.

Kise, K. & Doermann, D. (eds) (2007). *Proceedings of 2nd International Workshop on Camera-Based Document Analysis and Recognition*.

Kopec, G. (1997). Supervised template estimation for document image decoding, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **19**(12): 1313–1324.

Kuwano, H., Kurakake, S. & Odaka, K. (1997). Telop character extraction from video data, *Proceedings of Workshop on Document Image Analysis*, pp. 82–88.

Lienhart, R. & Wernicke, A. (2002). Localizing and segmenting text in images and videos, *IEEE Transactions on Circuits and Systems for Video Technology* **12**(4): 256–268.

Lyu, M., Song, J. & Cai, M. (2005). A comprehensive method for multilingual video text detection, localization, and extraction, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **15**(2): 243–255.

Mori, M., Sawaki, M. & Hagita, N. (2005). Video text recognition using category-dependent feature extraction based on feature compensation, *Systems and Computers in Japan* **36**(10): 1–8.

Mori, M., Sawaki, M., Hagita, N., Murase, H. & Mukawa, N. (2001). Robust feature extraction based on run-length compensation for degraded handwritten character recognition, *Proceedings of 6th International Conference on Document Analysis and Recognition*, pp. 650–654.

Mori, M., Sawaki, M. & Yamato, J. (2010). Robust character recognition using adaptive feature extraction method, *IEICE Transactions on Information and Systems* **E93-D**(1): 125–133.

Nakagawa, M., Yanagida, T., & Nagasaki, T. (1999). An off-line character recognition method employing model-dependent pattern normalization by an elastic membrane model, *Proceedings of 5th International Conference on Document Analysis and Recognition*, pp. 495–498.

Omachi, S., Sun, F. & Aso, H. (2000). A noise-adaptive discriminant function and its application to blurred machine-printed kanji recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(3): 314–319.

Sato, A. (2000). A learning method for definite canonicalization based on minimum classification error, *Proceedings of 15th International Conference on Pattern Recognition*, Vol. 2, pp. 199–202.

Sawaki, M. & Hagita, N. (1998). Text-line extraction and character recognition of document headlines with graphical designs using complementary similarity measure, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(10): 1103–1109.

Srihari, S., Hong, T. & Srikantan, G. (1997). Machine-printed japanese document recognition, *Pattern Recognition* **30**(8): 1301–1313.

Tsukumo, J. & Tanaka, H. (1988). Classification of handprinted chinese characters using nonlinear normalization and correlation methods, *Proceedings of 9th International Conference on Pattern Recognition*, Vol. 1, pp. 168–171.

Umeda, M. (1996). Advances in recognition methods for handwritten kanji characters, *IEICE Transactions on Information and Systems* **79**(5): 401–410.

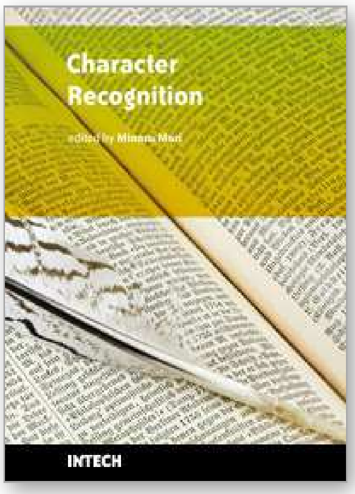
Wakahara, T. & Odaka, K. (1998). Adaptive normalization of handwritten characters using global/local affine transformation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20**(12): 1332–1341.

Xu, Y. & Nagy, G. (1999). Prototype extraction and adaptive OCR, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21**(12): 1280–1296.

Yamada, H., Yamamoto, K. & Saito, T. (1990). A nonlinear normalization method for handprinted kanji character recognition – line density equalization, *Pattern Recognition* **23**(9): 1023–1029.

Zhang, J. & Kasturi, R. (2008). Extraction of text objects in video documents: recent progress, *Proceedings of 8th International Workshop on Document Analysis Systems*, pp. 5–17.

Zhu, J., Hong, T. & Hull, J. (1997). Image-based keyword recognition in oriental language document images, *Pattern Recognition* **30**(8): 1293–1300.



Character Recognition
Edited by Minoru Mori

ISBN 978-953-307-105-3
Hard cover, 188 pages
Publisher Sciyo
Published online 17, August, 2010
Published in print edition August, 2010

Character recognition is one of the pattern recognition technologies that are most widely used in practical applications. This book presents recent advances that are relevant to character recognition, from technical topics such as image processing, feature extraction or classification, to new applications including human-computer interfaces. The goal of this book is to provide a reference source for academic research and for professionals working in the character recognition field.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821