Robust feature extraction based on run-length compensation for degraded handwritten character recognition

Minoru Mori, Minako Sawaki, Norihiro Hagita, Hiroshi Murase, and Naoki Mukawa

NTT Communication Science Laboratories, NTT Corporation
3-1, Morinosato-Wakamiya, Atsugi-shi, Kanagawa, 243-0198 Japan
mmori@eye.brl.ntt.co.jp

Abstract

Conventional features are robust for recognizing either deformed or degraded characters. This paper proposes a feature extraction method that is robust for both of them. Run-length compensation is introduced for extracting approximate directional run-lengths of strokes from degraded handwritten characters. This technique is applied to the conventional feature vector based on directional run-lengths. Experiments for handwritten characters with additive or subtractive noise show that the proposed feature is superior to conventional ones over a wide range of the degree of noise.

1. Introduction

In character recognition, we face two main problems: deformation and degradation. Conventional feature vectors are robust for either deformed characters or for degraded ones. For example, for overcoming degradation, Kopec [5], Xu et al. [14], Ho [4], and Sawaki et al. [11] have focused on designing reference templates that reflect the degraded condition and/or font styles. Sawaki et al. [10] have also proposed a robust discriminant function for recognizing characters with noise and texture. Unfortunately, these methods are effective only for machine-printed characters, since they employ image-based template matching.

Geometric and structural features are effective for recognizing multiple fonts and handwriting in the presence of deformation. In particular, the directional information of strokes is quite effective. The direction contributivity feature [1] and LSD [13, 15] are based on stroke run-length and are quite effective for handwriting and multi-fonts. Unfortunately, they are intolerant of character degradation because run-length extraction becomes inaccurate for degraded characters. To tackle this problem, preprocessing methods are often used for removing noise and filling white gaps [8, 7, 12]. However, these methods sometime eliminate crucial strokes or enhance noise. To counter the degradation, the Hough Transform has been used for line detection in noisy images [3, 6]. However, as it is based on voting, it does not ensure pixel continuity.

This paper, therefore, proposes a method that enables the extraction of stroke directional information from even degraded characters. We introduce a technique, called "run-length compensation " for extracting approximate directional run-lengths of strokes. It well counters the impact of additive and subtractive noise on stroke directional information. Experiments using ETL-9 handwritten Kanji character data show that the proposed method achieves much higher recognition rates than the conventional methods over a wide range of noise level.

2. Feature extraction for degraded character recognition

2.1. Run-length compensation

We assume that degradation comes from additive and subtractive noise. As the degree of these noise types increases, stroke run-length becomes harder to extract. Therefore, in place of run-length, we introduce "compensated run-length" which can better extract the stroke directional information from characters corrupted by noise. Run-length compensation utilizes the complementary relationship between additive and subtractive noise in terms of black and white runs and complexity. Additive noise has a complementary relationship to subtractive noise in terms of the four possible changes of black and white pixels (black-to-black, white-to-white, white-to-black, and black-to-white) that can occur along the scan direction of a character image. As additive noise increases, white runs become shorter while black runs hardly change or grow gradually; the complexity corresponding to changes of black and white pixels (white-to-black and black-to-white) increases. On the other hand,

0-7695-1263-1/01/$10.00 © 2001 IEEE

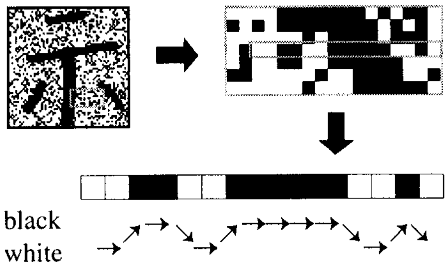


Figure 1. An example of extracting parameters from the input image.

as subtractive noise increases, black runs become shorter while white runs hardly change or grow gradually; complexity also increases.

Let G ($W \times W$ pixels) be an input binary image and G_w ($g_w(i,y)$: $I \times 1$ pixels; $i = 1, 2, \ldots, I$) be a rectangular window for extracting run-length. When horizontal scanning at y, the four possible changes (a, b, c, and e) at y are given by

$$a = \sum_{i=1}^{I-1} g_w(i,y) \cdot g_w(i+1,y), \quad (1)$$

$$b = \sum_{i=1}^{I-1} (1 - g_w(i,y)) \cdot g_w(i+1,y), \quad (2)$$

$$c = \sum_{i=1}^{I-1} g_w(i,y) \cdot (1 - g_w(i+1,y)), \quad (3)$$

$$e = \sum_{i=1}^{I-1} (1 - g_w(i,y)) \cdot (1 - g_w(i+1,y)). \quad (4)$$

Figure 1 shows an example of extracting parameters from a rectangular window in the input image.

Here we introduce run-length compensation. It's defined separately for additive and subtractive noise. That is, we assume that characters are degraded by either additive noise or subtractive noise. Compensated run-length r_t' is calculated as follows:

(1) In the case of additive noise

Let r_i, r_t, and r_n be the number of black pixels, run-length of stroke, and the number of noise pixels in the rectangular window, respectively. The relation among them is defined as

$$r_i = r_t + r_n. \quad (5)$$

Therefore, run-length r_t is given by

$$r_t = r_i - r_n = (1 - r_n/r_i) \cdot r_i. \quad (6)$$

As r_n/r_i expresses the percentage of noise in black pixels, it can be regarded as the degree of degradation. This value is equal to 0 for a noise-free image, while it approaches 1 as the image become more degraded.

Let the fluctuation from the mean on the noise-free data in terms of the number of black pixels be ΔK_a, and that from the mean in terms of the complexity be ΔV_a. β_a as the ratio of the fluctuation of the black pixel number to that of the complexity is thus given by

$$\beta_a = \Delta K_a / \Delta V_a. \quad (7)$$

β_a approaches 1 in the ideal condition and 0 as the image become degraded, because of the increase of complexity. Therefore, the relation between r_n/r_i and β_a is regarded as

$$r_n/r_i \fallingdotseq 1 - \beta_a. \quad (8)$$

Let $\bar{a}$, $\bar{b}$, $\bar{c}$, and $\bar{e}$ be the respective averages of a, b, c, and e calculated from noise-free training data. We define ΔK_a and ΔV_a as

$$\Delta K_a = (a + b)/(\bar{a} + \bar{b}),$$

$$\Delta V_a = (b + c)/(\bar{b} + \bar{c}). \quad (10)$$

Also, the number of black pixels r_i is given by

$$r_i = a + b. \quad (11)$$

Therefore, the compensated run-length r_t', an approximation of run-length r_t, is derived from Eq. (6) ~ (11) as

$$r_t' = \frac{(a + b)/(\bar{a} + \bar{b})}{(b + c)/(\bar{b} + \bar{c})} \cdot (a + b). \quad (12)$$

(2) In the case of subtractive noise

Under the same assumptions used in the case of additive noise, each value is defined as follows:

$$r_t = r_i + r_n = (1 + r_n/r_i) \cdot r_i, \quad (13)$$

$$\beta_s = \Delta K_s / \Delta V_s,$$

$$\Delta K_s = (e + c)/(\bar{e} + \bar{c}), \quad (15)$$

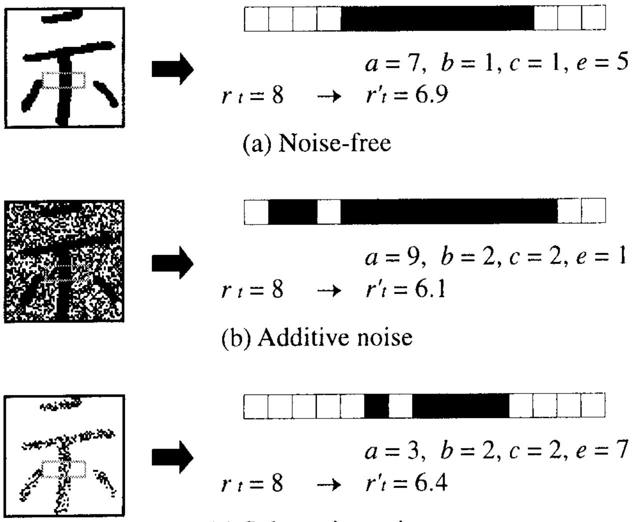
$$\Delta V_s = (b + c)/(\bar{b} + \bar{c}), \quad (16)$$

$$r_n/r_i \fallingdotseq 1 - \beta_s. \quad (17)$$

Therefore, the compensated run-length r_t' is given by

$$r_t' = \left(2 - \frac{(e + c)/(\bar{e} + \bar{c})}{(b + c)/(\bar{b} + \bar{c})}\right) \cdot (a + b). \quad (18)$$

For example, Figure 2 shows examples of the compensated run-length r_t' obtained from a rectangular window when $\bar{a}$, $\bar{b}$, $\bar{c}$, and $\bar{e}$ are 7.4, 0.9, 0.9, and 4.8, respectively. These r_t' take almost same values over different types of noise, and somewhat smaller than r_t.



(a) Noise-free

(b) Additive noise

(c) Subtractive noise

Figure 2. Examples of compensated run-length.

2.2. Detection of noise type

To detect noise type, we focus on the following projection value $p(y)$ [10] that originates from the global scan $(I = W)$:

$$p(y) = \frac{a \cdot e - b \cdot c}{\sqrt{(a+b)\cdot(c+e)\cdot(a+c)\cdot(b+e)}} \quad (19)$$

$$(-1 \leq p(y) \leq 1).$$

Since $a \cdot e$ decreases and $b \cdot c$ increases as additive noise increases, $p(y)$ can be used as a measure with which we can detect additive noise in the input images. On the other hand, as subtractive noise increases a decreases, e sees no change or slightly increases, and $b \cdot c$ increases. However, since the changes of these four parameters with subtractive noise are due to the degradation of character parts, the variation of $p(y)$ with subtractive noise is smaller than that seen with additive noise. the projection values $p(y)$ $(y = 1, 2, \ldots, W)$ for horizontal scanning and $p(x)$ $(x = 1, 2, \ldots, W)$ for vertical scanning are used as $2W$-dimentional feature vectors $\boldsymbol{T}$ for differentiating additive noise from subtractive noise.

Noise pattern Z_α is created by randomly setting a given $|\alpha|$ percentage of all black pixels to white for $\alpha < 0$ or all white pixels to black for $\alpha \geq 0$. The subtractive noise image X_α is formed by the AND-operation of X and Z_α in the case of $\alpha < 0$, while the additive noise pattern is yielded by the OR-operation of X and Z_α in the case of $\alpha \geq 0$.

In the learning stage, $2W$-dimensional feature vectors $\boldsymbol{T}$ are extracted for different $-70 \leq \alpha \leq 70$. Mean vector $\boldsymbol{M}$ for subtractive noise (or additive noise) are made of all $\boldsymbol{T}$ for $\alpha < 0$ (or $\alpha \geq 0$) from the training data in advance. Whether the input image includes additive or subtractive noise is determined by matching between $\boldsymbol{T}$ from input character image and $\boldsymbol{M}$ of additive or subtractive noise.

2.3. Feature vector based on compensated run-lengths

The direction contributivity feature [1] and LSD [13, 15] based on this feature have been proposed as a feature that reflects stroke directional information. Contributivity means the degree of contribution in terms of stroke direction. The direction contributivity feature is extracted by computing the following d_i $(i = 1, ..., 4)$:

$$d_i = l_i / \sqrt{\sum_{j=1}^{4} l_j^{\,2}} \quad (20)$$

where, l_1, l_2, l_3, and l_4 denote the run-lengths for the horizontal direction, vertical direction, and two diagonal directions, respectively.

Feature vectors based on compensated run-length r'_t are extracted as follows;

Step 1: An input image is divided into $N \times N$ local areas.

Step 2: On each black pixel, compensated run-length $r'_{t(i)}$ in the i $(i = 1, ..., 4)$ direction is calculated.

Step 3: Compensated run-lengths $r'_{t(i)}$ are averaged in each local area.

Step 4: Direction contributivity d_i $(i = 1, ..., 4)$ is calculated by substituting averaged $r'_{t(i)}$ for l_i in Eq. (20).

Step 5: d_i is multiplied by averaged $r'_{t(i)}$ in each direction in each local area.

The feature vector has $N \times N \times 4$ dimensions. Figure 3 shows examples of the proposed feature based on compensated run-lengths and the conventional LSD for the original image and the same image corrupted by additive noise. Figure 3 shows that our feature still retains stroke directional information for additive noise while LSD does not so.

3. Experiments

3.1. Data

We used the handwritten Kanji character database ETL-9 [9], which contains 3,036 categories. 100 samples per category were used as training data, and another 100 as test data. Additive and subtractive noise images were generated from training data as mentioned in **2.2** in the learning stage

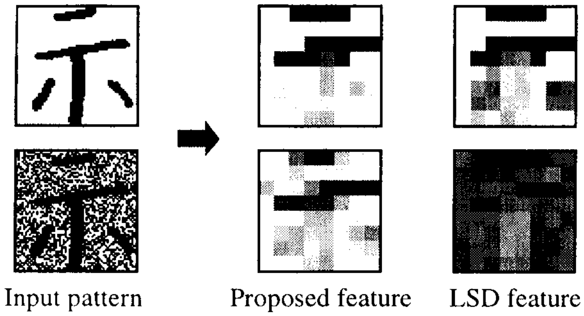


Figure 3. Examples of the proposed feature and LSD. Horizontal values are visualized.

for detecting noise type and from test data for evaluating the features in assessing the robustness against degradation. Noisy images with α percent noise were generated in the range of $-70 \leq \alpha \leq 70$.

3.2. Experimental results for detecting noise type

The detection of noise types was examined. $\boldsymbol{M}_\alpha$ with 15 levels of α were made in the range of $-70 \leq \alpha \leq 70$. $\boldsymbol{M}_{-70}$ through $\boldsymbol{M}_{-10}$ ($\boldsymbol{M}_{10}$ through $\boldsymbol{M}_{70}$) indicate subtractive (additive) noise. $\boldsymbol{M}_0$ indicates noise-free images. $3,036 \times 100 \times 15$ input noisy images were also generated from the original noise-free images. Euclidean distance was used as the discriminant function. Table 1 denotes the rates of noise type detection by matching the input image against M_α. These results show that our method achieves over 99.8% detection rate in the range of $-70 \leq \alpha \leq 70$ except $-10 \leq \alpha \leq 0$ for test data. The input images classified as $\alpha = 0$ (or $\alpha = -10$) are assigned to additive (subtractive) noise with 95.9% (94.1%) detection accuracy. As a result, the detection of noise type is not a problem.

3.3. Recognition results

The proposed feature based on compensated run-length was compared to LSD with/without 3×3 median filter. Each normalized pattern occupied 64×64 pixels. Here we focus on only the horizontal and vertical components in the direction contributivity while the original direction contributivity includes four directions involving two diagonal directions. This is because the purpose of this paper is to clarify that the feature based on run-length compensation preserves the recognition rate for noise free characters over a wide range of degradation rather than very high recognition accuracy. Therefore each feature vector consists of 128 dimensional components ($8 \times 8 \times 2$ directions). Euclidean distance was used as the discriminant function. From preliminary experiments, local window width was set at $I = 15$ pixels, which corresponds to ± 7 pixels from the pixel of interest. This width involves about 1 as the value of b and c (on average) in noise-free training data. Then $\bar{a}$, $\bar{b}$, $\bar{c}$, and $\bar{e}$ are 7.4, 0.9, 0.9, and 4.8, respectively.

Figure 4 plots the recognition accuracy of both features as functions of noise level α. It shows that the proposed feature is superior to conventional ones in the range of $-60 \leq \alpha \leq 70$. In particular, the proposed method better withstands additive noise.

The recognition rates decrease rapidly at heavy noise levels ($\alpha \leq -60$ or $\alpha \geq 60$). However, character images with these noise levels are hard to recognize even for humans. These results show the effectiveness of run-length compensation in resisting degradation. Run-length compensation can be applied to other feature vectors that reflect stroke directional information using run-lengths and will achieve high recognition accuracy for deformed characters, such as handwriting and multi-fonts.

4. Conclusions

We proposed a feature extraction method that allows the recognition of degraded and deformed characters. The technique of "run-length compensation" was introduced for extracting approximate stroke directional information even from degraded characters. It was applied to the conventional feature vector based on directional run-lengths. Experiments using ETL-9 handwritten Kanji character data show that "compensated run-length"-based feature vectors are superior to run-length based feature vectors over a wide range of noise degradation. The recognition rates will be improved by using the four directional components in the feature vectors from the property of the original feature vector [1]. When run-length compensation is also applied to a feature vector that is based on directional run-lengths, very high recognition rates are now possible for deformed a degraded characters.

Future works are to evaluate the recognition performance achieved by increasing the featurefs dimension through the use of diagonal directions too, and to apply our method to other types of degradation [2].

Acknowledgements We thank members of AIST (former ETL) of Japan for permitting the use of the ETL-9. Also we would like to acknowledge Dr. Kenichiro Ishii, director of our laboratories, for his support.

References

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

Table 1. Noise type detection accuracy [%].

Output	Input														
	Subtractive							Additive							
	-70	-60	-50	-40	-30	-20	-10	0	10	20	30	40	50	60	70
Subtractive	99.9	99.9	99.9	99.9	99.9	99.8	94.1	4.14	0.01	0	0	0	0	0	0
Additive	0.05	0.07	0.05	0.03	0.06	0.23	5.90	95.9	99.9	100	100	100	100	100	100

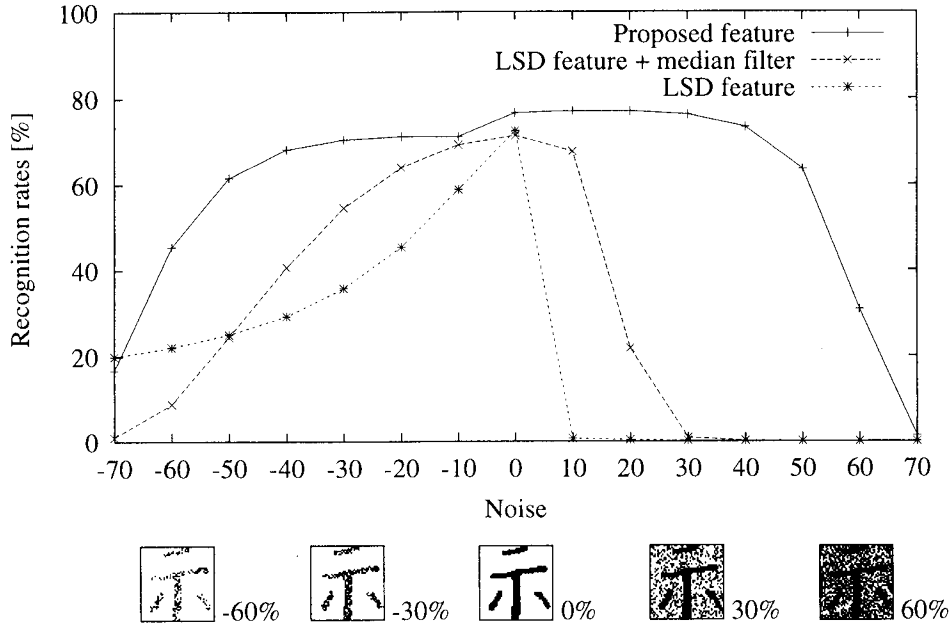


Figure 4. Recognition rates for patterns with subtractive or additive noise.

[2] H. S. Baird. Document image defect models. In H. S. Baird, H. Bunke, and K. Yamamoto, editors, *Structured Document Image Analysis*, pages 546–556. Springer-Verlag, 1992.

[3] O. Chutatape and L. Guo. A modified hough transform for line detection and its performance. *Pattern Recognition*, 32(2):181–192, Feb. 1999.

[4] T. K. Ho. Bootstrapping text recognition from stop words. In *Proc. of 14th ICPR*, pages 605–609, 1998.

[5] G. E. Kopec. Supervised template estimation for document image decoding. *IEEE Trans. Pattern Analysis & Machine Intelligence*, 19(12):1313–1324, Dec. 1997.

[6] J. W. Lee and I. S. Kweon. Extraction of line features in a noisy image. *Pattern Recognition*, 30(10):1651–1660, Oct. 1997.

[7] S. Liang and M. Ahmadi. A morphological approach to text string extraction from regular periodic overlapping text/background images. *CVGIP*, 56(5):402–413, Sept. 1994.

[8] H. Ozawa and T. Nakagawa. A character image enhancement method from characters with various background image. In *Proc. of 2nd ICDAR*, pages 58–61, 1993.

[9] T. Saito, H. Yamada, and K. Yamamoto. On the database ETL9 of handprinted characters in JIS Chinese characters and its analysis. *IECE Trans.*, J68-D(4):757–764, April 1985 (in Japanese).

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

[11] M. Sawaki, H. Murase, and N. Hagita. Character recognition in bookshelf images using context-based image templates. In *Proc. of 5th ICDAR*, pages 79–82, 1999.

[12] Z. Shi and V. Govindaraju. Character image enhancement by selective region-growing. *Pattern Recognition Letters*, 17(5):523–527, May 1996.

[13] S. N. Srihari, T. Hong, and G. Srikantan. Machine-printed japanese document recognition. *Pattern Recognition*, 30(8):1301–1313, Aug. 1997.

[14] Y. Xu and G. Nagy. Prototype extraction and adaptive ocr. *IEEE Trans. Pattern Analysis & Machine Intelligence*, 21(12):1280–1296, Dec. 1999.

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