Affine Invariant Information Embedment for Accurate Camera-Based Character Recognition

Shinichiro Omachi
Tohoku University, Sendai, 980-8579 Japan
machi@ecei.tohoku.ac.jp

Masakazu Iwamura
Osaka Prefecture University, Sakai, 599-8531 Japan
masa@cs.osakafu-u.ac.jp

Seiichi Uchida
Kyushu University, Fukuoka, 812-8581 Japan
uchida@is.kyushu-u.ac.jp

Koichi Kise
Osaka Prefecture University, Sakai, 599-8531 Japan
kise@cs.osakafu-u.ac.jp

Abstract

Recognizing characters in a scene image taken by a digital camera has been studied for decades. However, it is still a challenging problem to achieve high accuracy. In this paper, we propose a method of embedding information in a character pattern so that the class of the character can be identified. The information should be robust against geometric distortions since an image taken by a digital camera is usually geometrically distorted. In the proposed method, a character pattern is designed in two colors so that the information is embedded as the area ratio of regions of two colors. Since the area ratio is affine invariant, it is expected that the area ratio is correctly extracted even if a character image is affine-transformed. We generate character patterns with the embedded information and discuss the effectiveness of the proposed method.

1. Introduction

With the widespread use of digital cameras, recognizing characters in a scene image has been studied. However, it is still a challenging problem to recognize characters in a real scene image accurately because of low resolution, uneven lightning, geometric distortion, and so on [2, 5].

There are some methods for achieving highly-accurate recognition by designing special fonts [1]. Recently, a method for embedding a class information in a character pattern is proposed [6]. In the method, the pattern is designed with horizontal stripes so that the cross ratio of the stripe widths represents the class information. Since the cross ratio is projective invariant, theoretically the class information can be extracted correctly even if the character image is projectively distorted. However, it is not easy to extract cross ratios in high-accuracy from a real image because of noise or slight changes in the image.

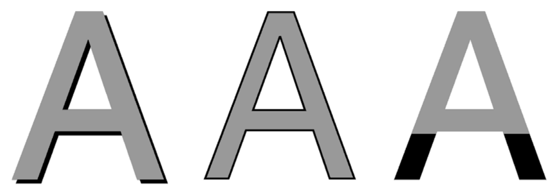


Figure 1. Various embedment methods.

Although an image taken by a digital camera is projectively distorted, it is thought that approximation by affine transformation is feasible if the target image is limited to a region of an individual character. Therefore, we focus on affine invariants. Parallel lines, ratio of lengths of parallel line segments, area ratio, and so on are known as affine invariants [3]. In this paper, we propose a method of designing a character pattern in two colors so that information is embedded as the area ratio of regions of two colors. It is expected that an area ratio is extracted correctly even if a character image is affine-transformed, since the area ratio is affine invariant. We generate character patterns with the embedded information and discuss the effectiveness of the proposed method.

2. Information embedment as area ratio

2.1. Embedment method

Information is embedded by designing a character pattern in two colors so that the area ratio of regions of two colors is a specific value. Since the only restriction is to use two colors, there are various embedment methods. Fig. 1 displays some examples.

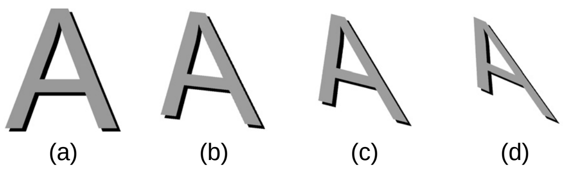


Figure 2. Affine-transformed images.

Table 1. Parameters of affine transformation and area ratio.

	(a)	(b)	(c)	(d)
a	1.0	0.9	0.8	0.7
b	0.0	0.1	0.2	0.3
c	0.0	0.1	0.2	0.3
d	1.0	0.9	0.8	0.7
ratio	0.2301	0.2300	0.2301	0.2300

It is commonly done to add shadow to a character pattern or to use different color for the profile line. The proposed method only changes the stroke width for these kinds of decorative characters. In addition, each character has its own natural area ratio since each character has its own shape. Therefore, a *natural* information embedment can be achieved by the proposed method.

2.2. Affine invariance

In general, an affine transformation maps a point $(x, y)^t$ to a point $(x', y')^t$ by the following equation.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}. \qquad (1)$$

In Eq. (1), e and f are parameters for translation, and do not take part in shape transformation of a pattern. Various geometric transformations can be performed by changing the values of a, b, c and d in Eq. (1).

Fig. 2 displays some examples. Fig. 2a shows the original image and Figs. 2b, 2c and 2d show transformed images. Shadow (black part) is added to a character pattern (gray part) of which height is 500 pixels. Parameters of affine transformation and area ratio of black and gray regions of these images are shown in Table 1. It is clarified from the table that the value of area ratio hardly changes even if the image is affine-transformed.

2.3. Example of character patterns

In this section, we show character patterns with class information. Twenty-six capital letters are used. As we mentioned in Section 2.1, it is thought that there is a natural value of area ratio for each character. In order to investigate

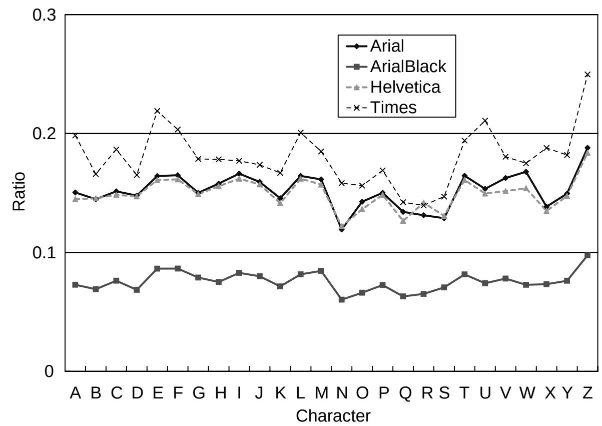


Figure 3. Area ratios of each character.

Table 2. Embedded values.

Character		A	B	C	D	E
Area ratio		0.33	0.25	0.32	0.27	0.44
F	G	H	I	J	K	L
0.43	0.31	0.34	0.40	0.36	0.26	0.42
M	N	O	P	Q	R	S
0.38	0.20	0.24	0.29	0.21	0.23	0.22
T	U	V	W	X	Y	Z
0.41	0.39	0.37	0.35	0.28	0.30	0.45

the natural values, area ratios are calculated for capital letters of four fonts, Arial, Arial Black, Helvetica, and Times. The embedment method shown in Fig. 1a is used. Given a character pattern of which height is 500 pixels, shadow is generated by moving the pattern right by ten pixels and below by ten pixels. Then the area ratio of the shadow and the character pattern is calculated. The results are shown in Fig. 3. The figure shows the tendency that if the area ratio of a character of a font is large, then the area ratio of the character of another font will be also large.

In order to realize a natural embedment, area ratios shown in Fig. 3 are averaged in each class, and the averaged values are sorted in ascending order. Then the value of $0.20 + 0.01(n - 1)$ is embedded in the characters of the nth class. The embedded values are shown in Table 2. The generated character patterns of two fonts of Helvetica and Times are shown in Fig. 4.

3. Experiment

3.1. Character recognition algorithm

In order to confirm the effectiveness of the proposed method, an experiment of recognizing the character patterns with class information is carried out. We use a simple char-

(a) Helvetica

(b) Times

Figure 4. Examples of generated character images.

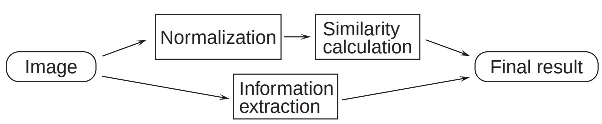


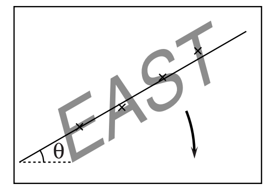
Figure 5. Character recognition algorithm.

acter recognition algorithm displayed in Fig. 5. A character pattern is normalized and similarities to the standard patterns of all the classes are calculated. Simultaneously, the embedded information is extracted. The results of similarity calculation and information extraction are combined and the final result is decided. Each process is described below.

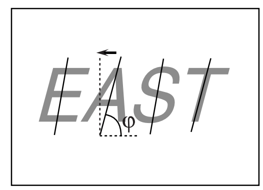
Normalization: In order to handle an affine-transformed image, we normalize the image. The normalization procedure is shown in Fig. 6. First we detect pixels that represent character patterns (Fig. 6a). Centroid is calculated for each connected component, and a line in Fig. 6b is determined by applying least squares method to the centroids. The inclination of the line is determined as the skew angle θ, and the skew is corrected by rotation (Fig. 6b). Next, for each row of each connected component, centroid is determined. Then a line is detected by applying least squares method to all the centroids for each connected component. The inclination of the line with the least error is determined as the slant angle φ, and the slant is corrected by shearing (Fig. 6c).

Similarity calculation: Normalized cross correlation (NCC) is used as similarity measure. A standard pattern is generated in advance by averaging character patterns for each class. NCC between the input pattern and standard pattern is calculated for each class. Ordered candidate classes are determined by sorting the NCCs in descending order.

(a) Input



(b) Skew correction



(c) Slant correction

(d) Result

Figure 6. Normalization.

Information extraction: Embedded information is extracted by calculating the ratio of the area of the character pattern and the area of the shadow. Difference between the extracted area ratio and the embedded value shown in Table 2 is calculated for each class. Ordered candidate classes are determined by sorting the differences in ascending order.

Combination of results: In order to determine the final result, the results of similarity calculation and information extraction are combined. If the skew and slant angle is correctly detected by the normalization process and the transformation of the pattern can be regarded as the affine transformation, it will not be difficult to recognize the image by the conventional character recognition method. Since the NCC will be large when the pattern is correctly recognized, we determine the final result as the candidate of the similarity calculation when the NCC is larger than a threshold t. If the NCC is smaller than or equal to the threshold t, a score of each class is calculated by summing the ranks of similarity calculation and information extraction. That is, if the class “A” ranks third in similarity calculation and ranks second in information extraction, the score will be five. The class with the smallest score is determined to be the final result. In the experiment, $t = 0.6$ is used.

3.2. Experimental results

Capital letters of four fonts of Arial, Arial Black, Helvetica and Times were used for the experiment. Class infor-

(a) (b)

Figure 7. Example of input image and normalized image.

Table 3. Camera data.

Speed	Diaphragm	Focal length
1/50	F10	34mm $\sim$ 45mm

mation was embedded in each character pattern in the manner described in Section 2.3. Character pattern is colored in red and shadow is colored in yellow. Twelve word patterns that were used for traffic signs (BEGIN, EAST, END, EXIT, LIMIT, NORTH, ONLY, ROAD, SOUTH, SPEED, STOP and WEST, 52 characters in total) were generated with the character patterns. Each word image is photographed in sunlight by a digital camera (Canon DS6041) with three camera angles. Camera data is shown in Table 3. The size of each image is 3072×2048, and approximate height of each character, skew angle and slant angle of the image are shown in Table 4. Each character pattern was recognized by the algorithm described in Section 3.1.

An example of an image taken by a digital camera and the normalized image is shown in Figs. 7a and 7b, respectively. In Fig. 7b, extracted character and shadow parts are only shown. Fig. 7b clarifies that the character patterns were not affine-transformed but actually projective-transformed.

The recognition accuracy is shown in Table 5. In the table, "conventional method" is the results without the information extraction. The effectiveness of the proposed method is confirmed since the accuracy has improved greatly for all the fonts. Especially the difference between the conventional method and the proposed method was large when the patterns are generated with Times font. Since character patterns of Times have complex structures, parameters of affine transformation were not correctly detected and the recognition accuracy of the conventional method was not so high. In this case the extracted class information worked effectively.

4. Conclusions

In this paper, we have proposed a method of designing a character pattern in two colors so that the information is embedded as the area ratio of regions of two colors. A natural information embedment can be achieved by the proposed method. We generated character patterns with class information and tested the effectiveness of the proposed method using a simple character recognition algorithm. The recognition accuracy of character patterns taken by a digital camera is greatly improved by the embedded information.

In this paper, we assigned information values in a simple manner shown in Table 2. Future work includes introducing the optimal information assignment method [4] into the proposed method, and improving the recognition algorithm to construct more accurate recognition system.

Table 4. Approximate height, skew angle and slant angle.

	Height [pixels]	Skew angle	Slant angle
Angle 1	450	25°	115°
Angle 2	550	0°	90°
Angle 3	450	-25°	75°

Table 5. Experimental results.

Font	Conventional method	Proposed method
Arial	91.7%	94.9%
Arial Black	87.2%	91.0%
Helvetica	91.0%	95.5%
Times	85.3%	92.3%
Average	88.8%	93.4%

References

[1] British Computer Society. *Character Recognition 1967*. Unwin Brothers Limited, London, 1966.

[2] D. Doermann, J. Liang, and H. Li. Progress in camera-based document image analysis. *Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)*, pages 606–616, August 2003.

[3] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, 2nd edition, 2004.

[4] M. Iwamura, S. Uchida, S. Omachi, and K. Kise. Recognition with supplementary information —How many bits are lacking for 100% recognition?—. *Proceedings of the First International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2005)*, pages 68–75, August 2005.

[5] J. Liang, D. Doermann, and H. Li. Camera-based analysis of text and documents: a survey. *International Journal on Document Analysis and Recognition*, 7(2-3):84–104, July 2005.

[6] S. Uchida, M. Iwamura, S. Omachi, and K. Kise. Data embedding for camera-based character recognition. *Proceedings of the First International Workshop on Camera-Based Document Analysis and Recognition (CBDAR2005)*, pages 60–67, August 2005.