2000 15th International Conference on Pattern Recognition

(V.4) (V.B)

Affine-Invariant Gray-Scale Character Recognition Using GAT Correlation

Toru Wakahara and Yoshimasa Kimura

NTT Cyber Solutions Laboratories
1-1 Hikari-no-oka, Yokosuka-shi, Kanagawa, 239-0847 Japan
E-mail: waka@marsh.hil.ntt.co.jp

Abstract

This paper describes a new technique of gray-scale character recognition that offers both noise-tolerance and affine-invariance. The key ideas are twofold. First is the use of normalized cross-correlation to realize noise-tolerance. Second is the application of global affine transformation (GAT) to the input image so as to achieve affine-invariant correlation with the target image. In particular, optimal GAT is efficiently determined by the successive iteration method. We demonstrate the high matching ability of the proposed method using gray-scale images of numerals subjected to random Gaussian noise and a wide range of affine transformation. The achieved recognition rate of 92.1% against rotation within 30 degrees, scale change within 30%, and translation within 20% of the character width is sufficiently high compared to the 42.0% offered by simple correlation.

1. Introduction

Most current OCR systems binarize the input image as preprocessing. However, input images often suffer from gray-scale degradation or image distortion and have complex backgrounds with color or textures. In such cases, the conventional binarization process loses a significant amount of information. This is a real problem because demand is increasing for the direct recognition of text in video frames and WWW images.

Regarding the direct recognition of gray-scale characters, there are two major approaches. The first approach is using topographic features [1]. This approach is effective in segmenting touching characters, but is sensitive to image defects and noise. The second approach is correlation-based matching [2]. However, correlation in itself is weak against geometrical distortion like affine transformation. On the other hand, the "perturbation method" [3] and "tangent-distance" [4] were proposed with the aim of distortion-tolerant image matching. However, both of these techniques can deal with only a limited range of affine transformation.

In our former paper [5], we introduced the concept of global affine transformation as a general deformation model to realize distortion-tolerant shape matching as applied to binary images of characters. By extending the concept of GAT as applied to the matching of gray-scale images, we proposed the promising technique of affine-invariant correlation of gray-scale characters using GAT iteration [6]. Conventional correlation-based matching was greatly reinforced in two ways. First is the use of normalized cross-correlation as a noise-tolerant matching measure. Second is the application of global affine transformation (GAT) to the input gray-scale image so as to realize affine-invariant correlation with the target gray-scale image.

This paper demonstrates the high matching ability of the proposed method using gray-scale images of numerals subjected to random Gaussian noise and a wide range of affine transformation including rotation, scale change, shearing, and translation. Moreover, extensive recognition experiments show that the proposed GAT correlation method achieves far superior recognition rates to the normalized cross-correlation without GAT and the conventional normalized inner product of the two images.

2. GAT correlation method

This paper deals with the matching of two gray-scale images. We denote the two images as **F**, the input gray-scale image, and **G**, the target gray-scale image, and represent **F** and **G** by gray level functions $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$, respectively, as follows:

$$\mathbf{F} = \{ f(\boldsymbol{r}) \},\ \mathbf{G} = \{ g(\boldsymbol{r}) \},\quad \boldsymbol{r} \in K \tag{1}$$

where $\boldsymbol{r}$ denotes a 2D loci vector defined in the bounded 2D domain of K. Of course, gray level functions $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$ take on non-negative values.

0-7695-0750-6/00 $10.00 © 2000 IEEE

2.1. Normalized cross-correlation

The conventional and most popular matching measure for the two images is the normalized inner product or "simple similarity measure" $S(f, g)$ defined by

$$S(f, g) \equiv (f, g) / \| f \| \cdot \| g \|$$
$$= \int_K f(\boldsymbol{r})\, g(\boldsymbol{r})\, d\boldsymbol{r} \,/\, [\, \int_K f(\boldsymbol{r})^2 d\boldsymbol{r} \ \int_K g(\boldsymbol{r})^2 d\boldsymbol{r}\,]^{1/2}, \quad (2)$$

where (f, g) denotes the inner product of f and g. However, the discrimination ability of the simple similarity measure $S(f, g)$ deteriorates considerably in the presence of image defects and noise [2].

As Iijima [7] proved theoretically, the normalized cross-correlation, defined by

$$C(f, g) = \int_K (f(\boldsymbol{r}) - \mu)\, (g(\boldsymbol{r}) - \nu)\, d\boldsymbol{r} \,/$$
$$[\, \int_K (f(\boldsymbol{r}) - \mu)^2 d\boldsymbol{r} \ \int_K (g(\boldsymbol{r}) - \nu)^2 d\boldsymbol{r}\,]^{1/2}, \quad (3)$$

where μ and ν are the mean gray levels of $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$, respectively, remains unchanged under image blurring or image degradation.

Therefore, we adopt the normalized cross-correlation to realize noise-tolerance. Moreover, we transform $f(\boldsymbol{r})$ and $g(\boldsymbol{r})$ so that they have zero mean and unit norm, and, finally, obtain the matching measure given by

$$C(f, g) = \int_K f(\boldsymbol{r})\, g(\boldsymbol{r})\, d\boldsymbol{r}. \quad (4)$$

However, we still have the problem that the correlation in itself cannot compensate for geometrical distortion such as affine transformation.

2.2. Affine-invariant correlation by GAT

This subsection introduces affine-invariant correlation by GAT and a simple computational model of determining optimal GAT using the successive iteration method [6].

First, we define global affine transformation (GAT). GAT is uniform affine transformation as applied to input gray-scale image **F** to generate GAT-superimposed input gray-scale image **F*** as follows

$$\mathbf{F}^* = \{ f^*(\boldsymbol{r}) \}, \quad \boldsymbol{r} \in K$$
$$\boldsymbol{r}^* = \mathrm{A}\boldsymbol{r} + \boldsymbol{b},$$
$$f^*(\boldsymbol{r}^*) = f^*(\mathrm{A}\boldsymbol{r} + \boldsymbol{b}) = f(\boldsymbol{r}), \quad (5)$$

where $\mathrm{A} = (\, \boldsymbol{a}_1\, \boldsymbol{a}_2 \,)$ is a 2×2 matrix representing rotation, scale change, and shearing; and $\boldsymbol{b} = (\, \mathrm{b}_x, \mathrm{b}_y \,)^T$ is a 2D translation vector. Here, $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ represent two basis vectors of the affine-transformed space.

Second, we define a fundamental objective function Φ of optimal GAT for affine-invariant correlation by

$$\Phi = \int_K f^*(\boldsymbol{r})\, g(\boldsymbol{r})\, d\boldsymbol{r} = \int_K f(\boldsymbol{r})\, g(\mathrm{A}\boldsymbol{r} + \boldsymbol{b})\, d\boldsymbol{r}$$
$$\rightarrow \quad \max \ \text{for } \mathrm{A}, \boldsymbol{b}, \quad (6)$$

which, however, requires exhaustive trial and error to determine optimal A and $\boldsymbol{b}$ because A and $\boldsymbol{b}$ are directly embedded in the general gray-level function g. Therefore, we adopt an equivalent objective function Ψ by introducing a Gaussian kernel of A and $\boldsymbol{b}$ into Φ as

$$\Psi = \iint_K f(\boldsymbol{r})\, g(\boldsymbol{r}')\, \exp(- \| \mathrm{A}\boldsymbol{r} + \boldsymbol{b} - \boldsymbol{r}' \|^2 / D)\, d\boldsymbol{r} d\boldsymbol{r}'$$
$$\rightarrow \quad \max \ \text{for } \mathrm{A}, \boldsymbol{b}, \quad (7)$$

where D specifies the spread of the Gaussian kernel. Also, it is to be noted that (7) is equivalent to (6) as $D \rightarrow 0$. Here, we give the D value in a deterministic way as

$$D = 1/2\, \{ \text{Mean of } [\, \min_{\boldsymbol{r}'} \| \boldsymbol{r} - \boldsymbol{r}' \|^2 ;$$
$$f(\boldsymbol{r}) = g(\boldsymbol{r}') \ \text{for } \forall \boldsymbol{r} \in K \,]$$
$$+ \text{Mean of } [\, \min_{\boldsymbol{r}} \| \boldsymbol{r}' - \boldsymbol{r} \|^2 ;$$
$$g(\boldsymbol{r}') = f(\boldsymbol{r}) \ \text{for } \forall \boldsymbol{r}' \in K \,]\, \}. \quad (8)$$

Third, by setting the differentials of Ψ with respect to A and $\boldsymbol{b}$ equal to zero and linearizing the obtained equations, we obtain the following set of linear equations:

$$\mathrm{O} = \iint_K \gamma(\boldsymbol{r}, \boldsymbol{r}')\, f(\boldsymbol{r})\, g(\boldsymbol{r}')\, \boldsymbol{r}\, (\mathrm{A}\boldsymbol{r} + \boldsymbol{b} - \boldsymbol{r}')^T$$
$$\times \exp(- \| \boldsymbol{r} - \boldsymbol{r}' \|^2 / D)\, d\boldsymbol{r} d\boldsymbol{r}',$$
$$\boldsymbol{0} = \iint_K \gamma(\boldsymbol{r}, \boldsymbol{r}')\, f(\boldsymbol{r})\, g(\boldsymbol{r}')\, (\mathrm{A}\boldsymbol{r} + \boldsymbol{b} - \boldsymbol{r}')$$
$$\times \exp(- \| \boldsymbol{r} - \boldsymbol{r}' \|^2 / D)\, d\boldsymbol{r} d\boldsymbol{r}', \quad (9)$$

where $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ is introduced as a matching stabilizing constraint based on topographic features as

$$\gamma(\boldsymbol{r}, \boldsymbol{r}') = \max \{\, (\nabla f(\boldsymbol{r}), \nabla g(\boldsymbol{r}')), 0 \,\}. \quad (10)$$

It is to be noted that $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ equals zero when the angle between two gradient vectors, $\nabla f(\boldsymbol{r})$ and $\nabla g(\boldsymbol{r}')$, is more than 90°. On the other hand, when the angle between the two gradient vectors is less than 90°, the greater the norms of $\nabla f(\boldsymbol{r})$ and $\nabla g(\boldsymbol{r}')$ are, the greater the value of $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ is. This can have the desired effect in that $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ reinforces the contribution of edges to correlation matching, while suppressing the contribution of grounds.

2.3. Successive iteration method for GAT

Start : Calculate the initial value of $C_0 = C(f, g)$ of (4) between the original input and target gray-scale images, **F** and **G**.

Loop : Determine the GAT components of A and $\boldsymbol{b}$ by solving the simultaneous linear equations of (9) by conventional techniques like Gaussian elimination [8]. Next, generate the GAT-superimposed input gray-scale image $\mathbf{F}^* = \{ f^*(\boldsymbol{r}) \}$ by (5) and substitute **F*** for **F**. Update the value of D of (8) and $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ of (10).

Pause : Calculate the updated value of $C_1 = C(f, g)$ of (4). If $C_1 > C_0$, substitute C_1 for C_0, and go to *Loop* ; otherwise, output the maximal value of C_0 as the final result of affine-invariant correlation, and then stop.

3. Experimental results

We conducted matching and recognition experiments using gray-scale images of numerals subjected to random Gaussian noise and a wide range of affine transformation.

Figure 1 shows all target gray-scale images $g(\boldsymbol{r})$ with zero mean and unit norm, where each point is denoted by black circle, double circle, white circle, dot, and blank according to temporary quantization into 5 discrete levels. The spatial resolution was 32×32. These images were free of noise, and served as clean templates.

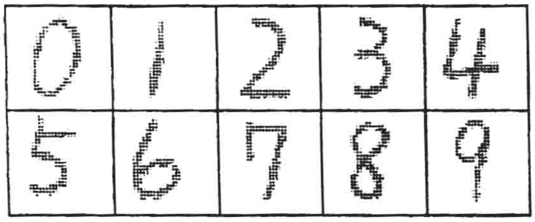


Figure 1. Target images of numerals.

On the other hand, each input gray-scale image $f(\boldsymbol{r})$ was artificially generated by applying arbitrary affine transformation A and $\boldsymbol{b}$ and, then, adding random Gaussian noise $\boldsymbol{n}(\boldsymbol{r})$ with zero mean and unit variance to one of the target images $g(\boldsymbol{r})$ as follows:

$$f(\boldsymbol{r}) = g(\mathrm{A}^{-1}(\boldsymbol{r} - \boldsymbol{b})) + \kappa \cdot \delta \cdot \boldsymbol{n}(\boldsymbol{r}), \quad \kappa \in (\,0, 1\,] \qquad (11)$$

where κ controls the magnitude of added random Gaussian noise, and δ denotes the dynamic range of $g(\boldsymbol{r})$. Finally, $f(\boldsymbol{r})$ was transformed to have zero mean and unit norm so that we calculate the normalized-cross correlation by (4).

Incidentally, when applying the GAT correlation method to these examples of digital images, we replaced the integral with the sum in (9) and calculated gray-scale gradients by the Roberts operator [9] in (10).

Figure 2 shows an example of the matching process using GAT iteration between the target image of "four" and the input image generated by the following affine parameters: $\|\boldsymbol{a}_1\| = 1.20$, $\|\boldsymbol{a}_2\| = 1.10$, $\angle \boldsymbol{a}_1 = -20°$, $\angle \boldsymbol{a}_2 = 60°$, $b_x = 2$, $b_y = -2$, and random Gaussian noise of $\kappa = 0.7$. Fig. 2 (a) shows the initial overlapped image of the target and input images, the correlation value is low at 0.282, where the points of the target image are denoted by asterisks. Fig. 2 (b)-(d) show GAT matching results at iterations of 3, 9, and 19, respectively, where the obtained correlation values were 0.394, 0.795, and 0.827 in this order. The estimated affine parameters for the result of Fig. 2 (d) were as follows: $\|\boldsymbol{a}'_1\| = 1.27$, $\|\boldsymbol{a}'_2\| = 1.30$, $\angle \boldsymbol{a}'_1 = -18.8°$, $\angle \boldsymbol{a}'_2 = 57.1°$, $b'_x = 2.48$, $b'_y = -1.48$, which were satisfactory compared to the correct ones.

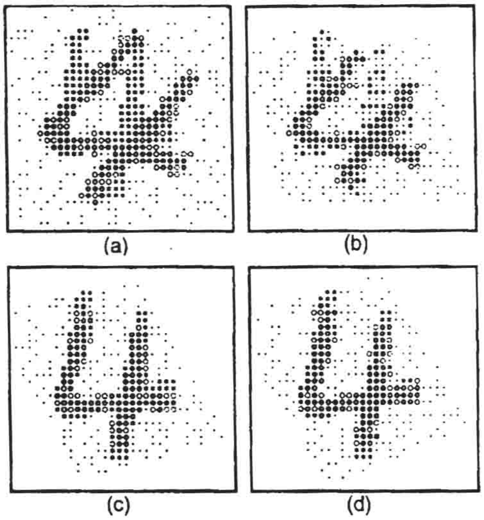


Figure 2. Example of GAT iteration.

Next, we show three kinds of quantitative results gained by applying the proposed GAT correlation method to the matching of input images artificially generated by either pure rotation or pure scale change or pure translation against their correct target images. Random Gaussian noise of $\kappa = 0.7$ was added to each input image.

First, Figure 3 shows the relation between the mean of normalized cross-correlation values and the rotation angle which was varied from −45° to +45° in 5 degree steps. Hence, the total number of input images was 10 digits times 19 rotation angles.

From Fig. 3, it is found that the converged correlation achieved a high value, more than 0.7, even if the original correlation value without GAT iteration was around 0.2.

Secondly, Figure 4 shows the relation between the mean of normalized cross-correlation values and the scale change which was varied from 0.5 to 1.5 in 0.1 steps. Here, the total number of input images was 10 digits times 11 scale changes.

From Fig. 4, it is found that the GAT correlation method is more robust against image expansion than against image shrinking. This asymmetry is due to blurring effect of the Gaussian kernel in (7) which is apt to bring about image contraction by GAT.

Thirdly, Figure 5 shows the relation between the mean of normalized cross-correlation values and the norm of translation, Δ, defined by $\Delta = \max(\,|b_x|, |b_y|\,)$, where b_x and b_y took integer values $\in [-5, +5]$. Hence, the total number of input images was 10 digits times 121 translations.

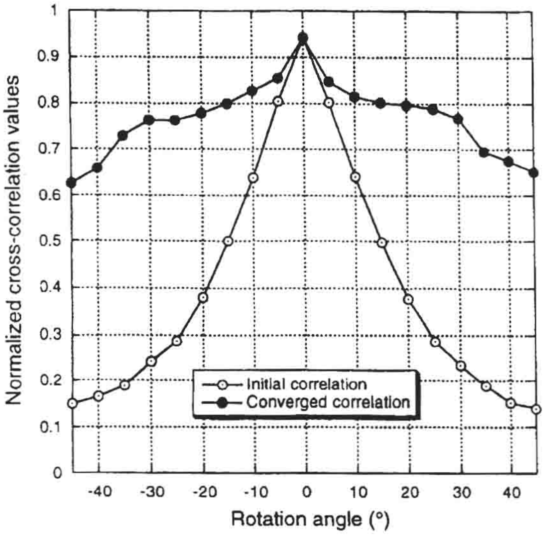


Figure 3. Relation between correlation values and rotation angle along with noise.

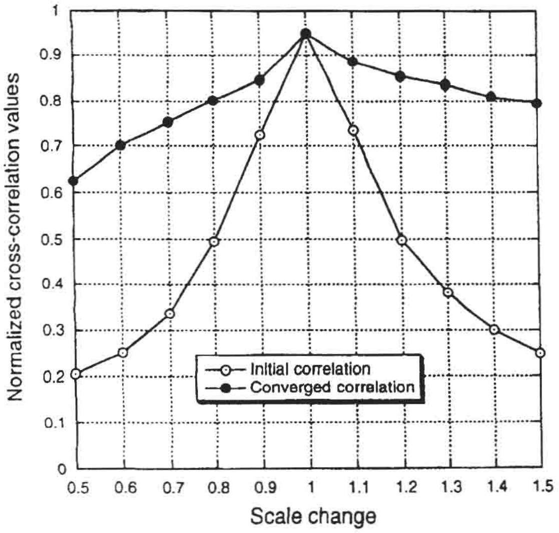


Figure 4. Relation between correlation values and scale change along with noise.

Figure 5 shows that the proposed method achieved a high correlation value, more than 0.8, even if the original correlation value without GAT iteration was less than 0.2.

From Fig. 3, Fig. 4, and Fig. 5, it is found that the proposed method is more robust against translation than against rotation or scale change. This is because the matching stabilizing factor $\gamma(\boldsymbol{r}, \boldsymbol{r}')$ of (10) requires high similarity in gray-scale gradients of matched points.

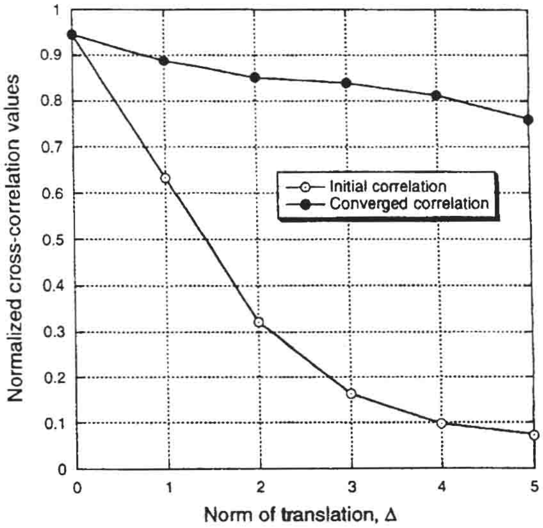


Figure 5. Relation between correlation values and norm of translation along with noise.

Finally, we show the results of extensive recognition experiments conducted using artificial input images subjected to combinations of rotation, scale change, and translation along with random Gaussian noise.

Figure 6 shows the relation between the recognition rates and the norm of translation, Δ, under various combinations of rotation and scale change. The range of rotation was set at within 30 degrees in 5 degree steps. Also, the range of scale change was set at between 0.7 and 1.3 in 0.1 steps. Hence, the number of artificially generated input images per digit for each translation vector $\boldsymbol{b}$ was 13 times 7. Finally, random Gaussian noise of $\kappa = 0.7$ was added to each input image.

In Fig. 6, the corresponding recognition rates obtained by using the normalized cross-correlation without GAT and the normalized inner product are also plotted. Here, we call these two conventional methods "simple correlation" in contrast to GAT correlation.

Figure 6 shows clearly that the discrimination ability of GAT correlation is far superior to that of simple correlation. Actually, the achieved recognition rate of 92.1% against rotation within 30 degrees, scale change within 30%, and translation of $\Delta \leq 3$ or within 20% of the character width is quite high compared to the 42.0% offered by simple correlation.

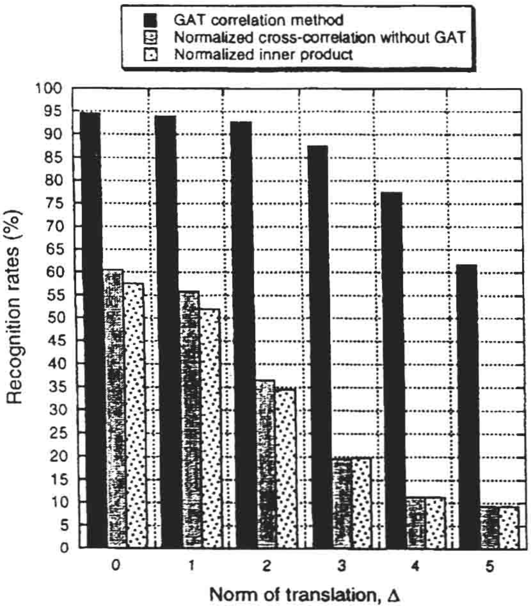


Figure 6. Relation between recognition rates and norm of translation under rotation, scale change, and noise.

A failure in GAT correlation can be attributed to any one of the following factors: 1) the "local optimum" problem in iteratively determining optimal GAT, 2) the excessive image contraction that occurs when the values of D of (8) are too large, and 3) the limit of topographic constraints using gray-scale gradients in $\gamma(r, r')$ of (10).

Finally, we discuss the trade-off between recognition time and recognition ability within GAT correlation.

If we adopt simple correlation by preparing multiple templates against rotation within 30 degrees in 5 degree steps, scale change between 0.7 and 1.3 in 0.1 steps, and translation of $\Delta \leq 3$ in integer steps, the total number of templates is 4,459=13×7×49. On the other hand, the present recognition time of GAT correlation is about 10^4 times larger than that of simple correlation using a single template. In this situation, simple correlation using 4,459 templates per digit is only about two times faster than GAT correlation. However, as is clear, if we prepare multiple templates against a wider range of affine transformation, including shearing, the recognition time of simple correlation would be much larger than that of GAT correlation. From this consideration, we can say that GAT correlation provides a very powerful and useful solution to the problem of affine-invariant gray-scale character recognition under both heavy image degradation and geometrical distortion.

4. Conclusion

We have demonstrated successful experiments in which the GAT correlation method was applied to the matching and recognition of gray-scale images of numerals subjected to random Gaussian noise and a wide variety of affine transformation. Future work is to apply our method to gray-scale character recognition not only in real-life paper documents but also in video frames and WWW images.

References

[1] L. Wang and T. Pavlidis, "Direct gray-scale extraction of features for character recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 15, pp. 1053-1067, 1993.

[2] M. Sawaki and N. Hagita, "Recognition of degraded machine-printed characters using a complementary similarity measure and error-correction learning," *IEICE Trans. Inf. & Syst.*, vol. E79-D, pp. 491-497, 1996.

[3] T. M. Ha and H. Bunke, "Off-line, handwritten numeral recognition by perturbation method," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, pp. 535-539, 1997.

[4] P. Simard, Y. LeCun, and J. Denker, "Efficient pattern recognition using a new transformation distance," *Advances in Neural Information Processing Systems*, vol. 5, pp. 50-58, Morgan Kaufmann, 1993.

[5] T. Wakahara and K. Odaka, "Adaptive normalization of handwritten characters using global/local affine transformation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 1332-1341, 1998.

[6] T. Wakahara and Y. Kimura, "Affine-invariant correlation of gray-scale characters using GAT iteration," *Proc. of 5th Int. Conf. Document Analysis and Recognition*, Sept. 1999, pp. 613-616.

[7] T. Iijima, *Pattern Recognition*. Tokyo: Corona, 1973, Chap. 6 (in Japanese).

[8] Mathematical Society of Japan, *Encyclopedic Dictionary of Mathematics*. Cambridge, MA: MIT Press, 1977.

[9] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. Second Edition. San Diego, CA: Academic Press, 1982.

Automatic Generation of Structured Hyperdocuments from Multi-Column Document Images *

Ji-Yeon Lee, Song-Ha Choi and Seong-Whan Lee
Center for Artificial Vision Research, Korea University
Anam-dong, Seongbuk-ku, Seoul 136-701, Korea
{jylee, shchoi, swlee}@image.korea.ac.kr

Abstract

In this paper, we propose two methods for converting complex multi-column document images into HTML documents, and a method for generating a structured table of contents(ToC) page based on the logical structure analysis of the document image.

Experiments with various kinds of multi-column document images show that HTML documents corresponding to the paper documents can be generated in a visual layout, and that their structured table of contents page, with the hierarchically ordered section titles hyperlinked to the contents, can be also produced by the proposed methods.

1. Introduction

The popular use of the internet, in these days, demands a type of documents accessible through the Web, for the purpose of sharing the documents. However, only a few works have been done on the conversion of paper documents into hyperdocuments. Most of the studies were about the conversion of single column document images that include text and image objects only [4, 5].

In this paper, we propose a system that converts multi-column document images into HTML documents, and also generates a table of contents page for the converted HTML documents. For the system, two methods are presented; one is implemented using the table structure and the other using their layer structure. We also suggest a method of generating a ToC page through a logical structure analysis [2].

2. HTML conversion of multi-column document images

2.1. An approach based on table structure

As a result of the geometrical structure analysis, a document image is divided into objects, each of which is classified as text, image, or table object. It is very easy to represent text and image objects by inserting simple tags in an HTML document without any specialized operations. On the other hand, table objects, having various formats, need some manipulations for conversion.

We propose a new algorithm for converting table objects in a paper document image into table objects in HTML format and apply this to the conversion of a multi-column document image into a hyperdocument as it is.

(1) Object merging

To have the objects fit into a virtual table format, we merge them and modify their coordinates, as shown in Figure 1. First, we divide a document image horizontally into regions where the value of a horizontal projection profile is zero. Second, for each of the horizontally divided regions, we divide it vertically where the value of a vertical projection profile is zero. Finally, from left to right and from top to bottom, object merging takes place in each region.

(2) Object ordering

By regarding each of the merged objects as a cell in a table, we construct a virtual table and arrange the objects in the same order that the cells of a table are created in an HTML document. The criteria for the arrangement order of the objects are as follows:

Let O_i and O_j be the *i*th and *j*th objects, respectively. Let x_i^{TL} and y_i^{TL} be the top and leftmost *x* and *y* coordinates of the O_i.

(1) If $|y_i^{TL} - y_j^{TL}| < th$ and $x_i^{TL} < x_j^{TL}$, $i, j = 1,...,n$, then O_i and O_j have the following properties:

- O_i and O_j exist in the same row.
- O_i has priority over O_j.

(2) If $|y_i^{TL} - y_j^{TL}| > th$ and $y_i^{TL} < y_j^{TL}$, $i, j = 1,...,n$, then O_i and O_j have the following properties:

- O_i and O_j exist in different rows.
- O_i has priority over O_j.

Figure 2 shows the result images of each stage of the approach based on the table structure.

*This research was supported by Creative Research Initiatives of the Ministry of Science and Technology, Korea.

0-7695-0750-6/00 $10.00 © 2000 IEEE

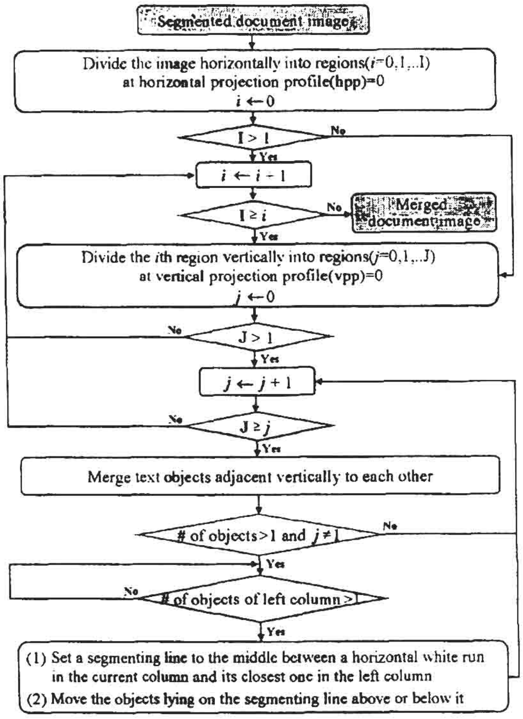

Segmented document image
Divide the image horizontally into regions(i=0,1,..I) at horizontal projection profile(hpp)=0 i ← 0
I > 1
No
Yes
i ← i + 1
I ≥ i
No
Merged document image
Yes
Divide the ith region vertically into regions(j=0,1,..J) at vertical projection profile(vpp)=0 j ← 0
No
J > 1
Yes
j ← j + 1
No
J ≥ j
Yes
Merge text objects adjacent vertically to each other
# of objects>1 and j≠1
No
Yes
No
# of objects of left column>1
Yes
(1) Set a segmenting line to the middle between a horizontal white run in the current column and its closest one in the left column
(2) Move the objects lying on the segmenting line above or below it

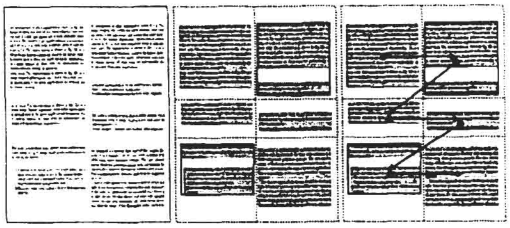
Figure 1. Flowchart of merging text objects

(3) Table-to-HTML algorithm

In this section, we present a Table-to-HTML algorithm as follows. The algorithm converts not only a table object in a document image into that of its HTML document but also a multi-column document image into its corresponding HTML document by considering it as a table object as a whole.

```
Initialize: $Colspan = Rowspan = 1, i = j = 0$;
If $Cell[i][j]_T$ and $Cell[i][j]_L == 1$, then {
    (1)  while ($Cell[i][j]_R == 1$)
             Increase $Colspan$ and $j$ values by one,
    (2)  while ($Cell[i][j]_B == 1$)
             Increase $Rowspan$ and $i$ values by one
}
Else move to next cell
```

$Cell[i][j]_T$, $Cell[i][j]_B$, $Cell[i][j]_L$ and $Cell[i][j]_R$ are the top, bottom, left and right lines of the cell, located in the *i*th row and the *j*th column of a table, respectively. While each cell is checked from left to right and from top to bottom, tagging is performed. And if a new row starts, we insert <TR><TD> tags; otherwise, we insert only a <TD> tag.



(a) Original document (b) Merging (c) Ordering

Figure 2. Result images of each conversion stage based on table structure

2.2. An approach based on layer structure

In this section, we use the layer structure for the conversion of multi-column document images.

(1) Object resizing

In order to have the converted HTML document fit into the screen's size regardless of the size of the input document, the objects in the image need to be resized in a constant rate before conversion. The conversion rate($CR = P_w/S_w$) for resizing is defined as the ratio of the width of the input image(P_w) to the width resolution of the user's screen(S_w).

This rate is also applied to the conversion of font size and line space as follows.

- $\tilde{O}_{fs} = O_{fs} \times CR$
  where O_{fs} and $\tilde{O}_{fs}$ are the font sizes of a text object and its converted object, respectively.
- $\tilde{O}_{ls} = (\tilde{O}_h - \tilde{O}_{fs} \times TL_n)/(TL_n\text{-}1)$
  where $\tilde{O}_{ls}$, $\tilde{O}_h$, and TL_n are the converted line space, the height of a text object, and the number of the lines in the text object, respectively.

(2) The structural properties of a text object

For more exact conversion, we must know concrete properties of the objects. In the discussion on the properties of text objects, text lines can be largely classified into 4 types under the following conditions.

Let $O_{sx(ex)}$ be the starting(ending) x coordinate of an object and $TL_{sx(ex)}$ be the starting(ending) x coordinate of a text line. Then,

- *Indented Line(IL)*, if $(\Delta sx>th) \wedge (\Delta ex<th)$
- *Entered Line(EL)*, if $(\Delta sx<th) \wedge (\Delta ex>th)$
- *New Line(NeL)*, if $(\Delta sx>th) \wedge (\Delta ex>th)$
- *Normal Line(NoL)*, if $(\Delta sx<th) \wedge (\Delta ex<th)$

where Δsx and Δex represent $|O_{sx}\text{-}TL_{sx}|$ and $|O_{ex}\text{-}TL_{ex}|$, respectively.

According to the line types defined above, proper tags are inserted to the HTML document. When a TL is of IL

type, "text-indent:Npx;" is inserted. When it is of EL type, "<BR>" is inserted. When it is of NeL type, "<BR>" and "text-indent:Npx;" are inserted. Here, N represents an indented distance in pixel.

3. Generation of a structured hyperdocument based on logical structure analysis

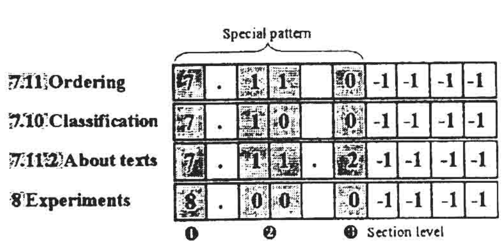
In this section, we take many related papers as input, and automatically generate a ToC page by extracting section titles and arranging them with respect to their hierarchical relation. A ToC page generated based on the logical structure analysis provides us with a clear overview of the logical flow of the entire input documents and its hyperlinks to the contents makes feasible the retrieval by section title or page number.

3.1. Section title extraction

We convert a technical paper that includes various objects and has well-formatted structure. For generation of a ToC page, we extract only section titles from the objects by referring to text line types defined in Section 2.2. When the current text line is of any type and starts with a special pattern and the previous line is of *EL* type, we regard the current line as a section title candidate. What is meant by the ***special pattern*** is the pattern in the form of [(number or text)+symbol(including space)]. Figure 3(a) shows an example of section title candidates extracted from a page of a technical paper.

3.2. Section number sorting and verification

We modify the pattern of a section title candidate as shown in Figure 3(b) and, starting at the highest level, sort all section title candidates by section number in ascending order at each level. We construct a hierarchical tree of the sorted section numbers. To verify their suitability, the section numbers at each level of the tree are searched for any missing or inappropriate section numbers.



7.11.Ordering	7	.	1	1		0	-1	-1	-1	-1
7.10.Classification	7	.	1	0		0	-1	-1	-1	-1
7.11.2.About texts	7	.	1	1	.	2	-1	-1	-1	-1
8.Experiments	8	.	0	0		0	-1	-1	-1	-1

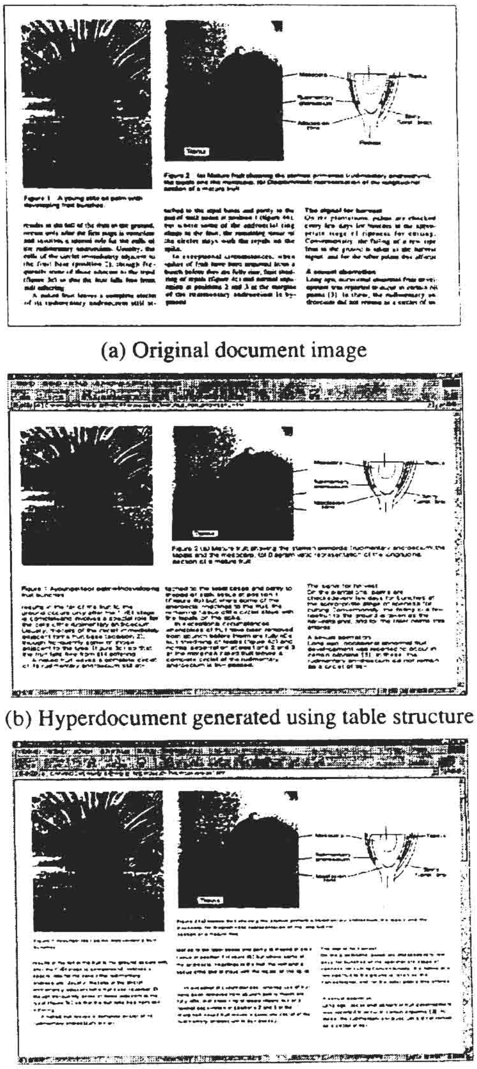
(a) Original pattern (b) Modified pattern for sorting

Figure 3. Section title candidates

4. Experimental results and analysis

Experiments were carried out with various kinds of the document images taken from magazines, newspapers, books, scientific and technical journals, manuals, UWDB(the database of University of Washington) [3], *etc.*

Figure 4(b) and Figure 4(c) show an example of the converted images using the table and layer structure, respectively. As a result of using these two approaches, HTML documents are generated very similar to the original document images both in appearance and logic.

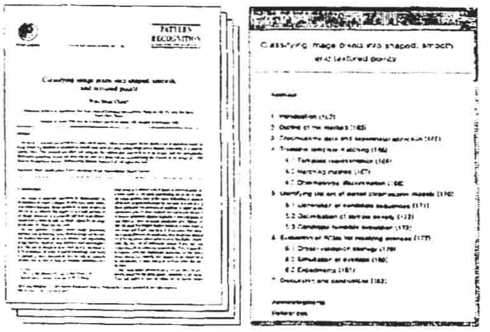


(a) Original document image

(b) Hyperdocument generated using table structure

(c) Hyperdocument generated using layer structure

Figure 4. Example of the converted images



(a) Input document images (b) Table of contents page

Figure 5. Structured ToC generation

In case of the conversion based on the table structure, the sizes of spaces between the objects in an HTML document may be considerably different from one another. This is because the objects are merged for arrangement in a table format convertible into an HTML document. Besides, an object lying across columns cannot be represented, although a partial overlap of objects in a column can be avoided. Therefore, the conversion based on the table structure is adequate for conversion of formatted document images like general books and technical papers. On the other hand, in the case of using the layer structure, the conversion of complex and unformatted multi-column document images, *i.e.*, documents without regular shape or structure, like magazines, advertisements, *etc.*, is performed well. However, several objects may overlap when the font size or line space of a text object is not correctly calculated, and the spaces between the objects may become larger or smaller than those of the objects in the original image.

Figure 5(b) shows a ToC page generated from input document images, Figure 5(a). Each section title is arranged hierarchically and has the page number of the beginning of the section and is hyperlinked to it.

Table 1 shows the experiment result of extracting section titles from different kinds of papers. N_c, N_{fn} and N_{fp} denote the number of section titles correctly extracted, false negatives, and false positives, respectively. In X/Y representation of N_c, X and Y denote the number of the correctly extracted section titles and the total number of the section titles in the input documents, respectively. As shown in Table 1, a ToC was generated with accuracy of about 90%.

Table 1. Results of section title extraction

Papers		Section title extraction		
		N_{fn}	N_{fp}	N_c
Journal	PR [1]	20	13	116/137 (85%)
	PAMI [2]	19	11	142/158 (90%)
Conference	ICPR [3]	8	7	75/81 (92%)
	ICDAR [4]	7	8	102/115 (89%)

5. Conclusions and further research

In this paper, we proposed two methods for converting multi-column document images into HTML documents using the table and layer structures and also proposed a method for generating a structured ToC page by extracting the section titles from input document images.

For the conversion of each paper image into its hyperdocument, the proposed conversion methods were tested on various kinds of complex multi-column document images. Experimental results revealed that the proposed methods performed well for various kinds of multi-column document images. However, they showed different performance on different types of document images. Hence, a scheme is needed to determine which one of the conversion methods to be applied for better performance when document images are given.

For the generation of a structured ToC page, experiments were carried out on technical papers. Experimental results showed that, using the proposed ToC generation method, we could create it by extracting the section titles from input document images without regard to the accuracy of character recognition.

References

[1] T. G. Kieninger and A. Dengel. A paper-to-html table converting system. *Proc. of the 3rd IAPR Workshop on Document Analysis Systems, Nagano, Japan*, pages 356–365, Nov. 4-6, 1998.

[2] C. Lin, Y. Niwa, and S. Narita. Logical structure analysis of book document images using contents information. *Proc. of the 4th ICDAR, Ulm, Germany*, pages 1048–1054, Aug. 18-20, 1997.

[3] I. Phillips, S. Chen, and R. Haralick. Cd-rom document database standard. *Proc. of the 2nd ICDAR, Tsukuba, Japan*, pages 478–483, Oct. 20-22, 1993.

[4] T. Tanaka and S. Tsuruoka. Table form document understanding using node classification method and html document generation. *Proc. of the 3rd IAPR Workshop on Document Analysis Systems, Nagano, Japan*, pages 157–158, Nov. 4-6, 1998.

[5] M. Worring and A. W. M. Smeulders. Content based internet access to paper documents. *International Journal on Document Analysis and Recognition*, 1(4):209–220, 1999.

[1] Pattern Recognition, Vol. 31, No. 4, April 1998.

[2] IEEE Trans. on PAMI, Vol. 20, No. 12, Dec. 1998.

[3] Proc. of Int. Conf. on Pattern Recognition, 1998, pp. 949-984.

[4] Proc. of Int. Conf. on Document Anal. and Recog., 1997, pp. 1-50.

Automatic Quality Measurement of Gray-Scale Handwriting Based on Extended Average Entropy*

Jeong-Seon Park, Hee-Joong Kang and Seong-Whan Lee
Center for Artificial Vision Research, Korea University,
Anam-dong, Seongbuk-ku, Seoul 136-701, Korea
{jspark, hjkang, swlee}@image.korea.ac.kr

Abstract

With a surge of interest in OCR in 1990s, a large number of handwriting or handprinting databases have been built one after another around the world. One problem that researches encounter today is that all the databases differ in various ways including the script qualities.

This paper proposes a method for measuring handwriting qualities that can be used for comparison of databases and objective test for character recognizers. The key idea involved is classifying character samples into a number of groups each characterizing a set of qualities.

In order to evaluate the proposed method, we carried out experiments on KU-1 database. The result we achieve is meaningful and the method is helpful for the target tasks.

1. Introduction

There are several handwritten character databases[5], covering a variety of writing styles and shape distortions, which have been frequently used to test the performance of various recognition systems. In order to evaluate and compare objectively the performance of those systems, it is necessary to carry out the task on common databases. However, this is often not possible, because of the unique properties of character databases each of which is used for and adapted to training and testing each specific recognition method. It is, therefore, desirable to estimate the level of recognition difficulty of a given database by some automatic quality measurement which is independent of the characteristics of recognition methods.

To date, several methods have been reported by concerning the evaluation of handwriting qualities[1, 2, 3, 4]. In the human perception based method[3], however, the use of the evaluation measures which are computed by subjective factors to determine the weight coefficient of the quality evaluation measure did not guarantee an objective evaluation of handwriting qualities. In the recognition based method[1], the sorted quality of each character is strongly affected by the features of the specified recognition system, also the simple, iterative method could not set the evaluation reference correctly. Finally, entropy based methods[2, 4] only evaluated the degree of variation among the data of the same character.

In this paper, we propose an *extended average entropy(EAE)* measure which has been stemmed from the *average entropy(AE)* in binary-scale[2], to directly measure the handwriting variations in gray-scale databases without adopting any error-causing binarization. Then, we use the EAE measure to evaluate the quality of handwriting and to classify all samples within a class into several groups according to their handwriting qualities.

2. Variation measures

In this section, we introduce the *average entropy(AE)* measure in binary-scale and extended the AE measure to gray-scale. The *extended average entropy(EAE)* will be used in next section, to measure the handwriting qualities of each group.

2.1. Average entropy in binary-scale

Given a collection of M binary images of $X \times Y$ pixels for a class, let $f(x,y)$ be the number of black pixels at position (x,y) for the collection. Then we can estimate the probability of black pixel occurring at position (x,y) as follows:

$$p(x,y) = \frac{f(x,y)}{M}, \quad \begin{cases} x = 1,\ 2,\ \cdots,\ X \\ y = 1,\ 2,\ \cdots,\ Y \end{cases} \qquad (1)$$

where X, Y are the horizontal/vertical dimension respectively, and M is the number of sample images in a class.

Then, the entropy at position (x,y) is calculated by the following equation[2]:

$$\begin{aligned} h(x,y) &= -p(x,y)\log_2 p(x,y) \\ &\quad -(1-p(x,y))\log_2(1-p(x,y)), \end{aligned} \qquad (2)$$

[*] This research was supported by Creative Research Initiatives of the Ministry of Science and Technology, Korea.

0-7695-0750-6/00 $10.00 © 2000 IEEE

where $(1 - p(x, y))$ denotes the probability of a white pixel occurring at position (x, y) in the set of binary-scale images. The entropy has a value of $0 \leq h(x, y) \leq 1$. Using Eq. (2), we defined the *average entropy(AE)* of the collection of class images, e^a, as follows:

$$e^a = \frac{1}{X \cdot Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} h(x, y), \tag{3}$$

where the *AE* has a value of $0 \leq e^a \leq 1$. $e^a = 0$ when all images in a class are the same, and $e^a = 1$ when the probability of a black pixel occurring at every position is exactly $\frac{1}{2}$.

2.2. Extended average entropy in gray-scale

Given M images of size $X \times Y$ with L gray levels $l = \{0, 1, 2, \cdots, L-1\}$, let the gray level of a pixel (x, y) be denoted by $I(x, y)$. Then, we define the frequency of the gray level l at position (x, y) as follows:

$$F(x, y; l) = \sum_{m=1}^{M} C_m(x, y; l), \tag{4}$$

where $C_m(x, y; l) = 1$ when $I(x, y) = l$, i.e., the gray level at position (x, y) is equal to l. Therefore, the frequency has a value of $0 \leq F(x, y; l) \leq M$.

The probability of the gray level l at position (x, y) can be easily calculated by

$$P(x, y; l) = \frac{F(x, y; l)}{M}, \quad \begin{cases} x = 1,\ 2,\ \cdots,\ X \\ y = 1,\ 2,\ \cdots,\ Y \\ l = 0,\ 1,\ \cdots,\ L-1 \end{cases} \tag{5}$$

with the following stochastic constraint:

$$0 \leq P(x, y; l) \leq 1, \quad \sum_{l=0}^{L-1} P(x, y; l) = 1.$$

We define the entropy at position (x, y) in gray-scale images of a class as follows:

$$H(x, y) = -\sum_{l=0}^{L-1} P(x, y; l) \log_L P(x, y; l). \tag{6}$$

Note that the base of the logarithm differs from that of the entropy in binary-scale.

Using Eq. (6), we define the *extended average entropy (EAE)* in gray-scale, E^A, as follows:

$$E^A = \frac{1}{X \cdot Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} H(x, y), \tag{7}$$

where the EAE has a value of $0 \leq E^A \leq 1$ with the equalities in order when all images in a class are the same and when the probabilities $P(x, y; l)$ for all l at each position are the same.

3. Automatic handwriting quality measurements

An overview of the proposed method is shown in Figure 1. As can be seen from the figure, no preprocessing and feature extraction is used because our intention is to measure the handwriting variation of original data only.

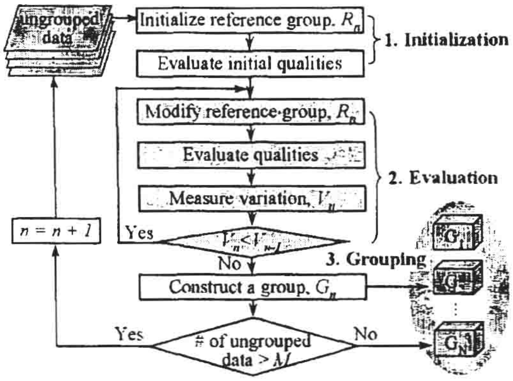


Figure 1. Overview of the proposed method

3.1. Initialization step

There are many types of variations in handwritten characters, including different writing styles, obviously it is helpful to start with a good reference group for a successful grouping of character samples.

In this phase, for making the n-th group G_n, the temporary reference group is set and all data are sorted according to the results of initial quality evaluation.

- **Initialize a reference group, R_n** : Select a set of M arbitrary samples from the ungrouped set, and assign it to the temporary reference group R_n. Then, the corresponding reference template is created by the average behavior of the samples in the reference group as follows:

$$A_n(x, y) = \frac{1}{M} \sum_{m=1}^{M} I_m(x, y), \tag{8}$$

  where M is the number of gray-scale images in the reference group. Then, the average image has the value of $0 \leq A_n(x, y) \leq L - 1$ for all (x, y).
- **Evaluate initial quality** : The qualities of all samples in the ungrouped set are evaluated and sorted according to their distance from the reference template.

  Let I_i be the i-th gray-scale image in the ungrouped set and A_n is the average image of gray-scale images of a current reference group as defined in Eq. (8). Then, $D_i = ||I_i - A_n||, (i = 1, 2, \cdots, M)$ is the distance of the i-th image to the reference template.

3.2. Evaluation step

Since the initialization of the the reference group by the previous method[1] is too simple, there may be unnecessary computation and the good reference group might not be set. Therefore, the proposed method examines the variation within the reference group, to confirm its suitability. In this phase, the reference group is modified and the quality of each sample is evaluated and the variation measure of the group is examined to guarantee that the reference group is properly set.

- **Modify reference group, $\mathbf{R}_n$** : The reference group R_n is replaced by the first M samples of the sorted data, and a reference template is made by the average image using Eq. (8).
- **Evaluate qualities** : The quality of each sample is evaluated and sorted according to its distance from the reference template the same as in the initialization step.
- **Measure variation, V_n** : In order to ensure that the reference group is good, the EAE in gray-scale is calculated as $V_n = E^A$. If the variation is less than the previous variation value, V_{n-1}, the reference group is regarded as good group but still have the possibility to be improved by the reference group modification.

3.3. Grouping step

Each group is constructed by assigning a different number of samples according to their EAE measure, i.e. the greater the EAE, the smaller the number of samples in the group; and the smaller the EAE, the larger the number of samples. Based on this concept, we construct the group G_n using the following procedure:

```
while ($V_n < V_T$) {
    add the next similar image to the current group, $G_n$
    calculate the handwriting variation, $V_n$
}
remove the grouped data from the ungrouped data
```

By applying this procedure, we expect that the handwriting samples are grouped with a different number of data according to their handwriting variations.

Figure 2 shows an example of the process constructing the fourth group of Hangul character '완'. In the figure, upper images are the reference templates made by averaging samples of the reference group, and lower images are some samples of constructed to the fourth group G_4, while n denotes the iteration count and V_n denotes the variation value. As the iteration count increases, the reference template has better shape and the variation values are reduced and both are converged after 6 times of iteration.

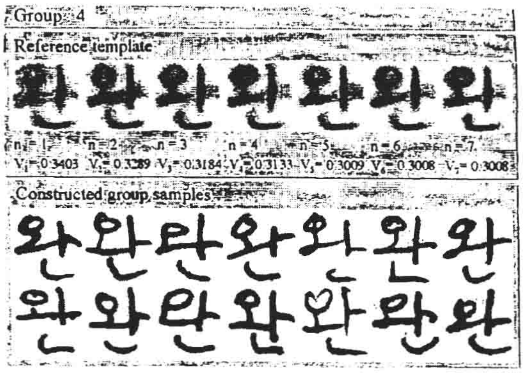


Figure 2. An example sequence of constructing the fourth group of character '완'

4. Experimental results and analysis

4.1. The KU-1 database

As detailed in previous research[5], KU-1 is a large-set off-line handwritten Hangul character database. It consists of 1,000 sets of the 1,500 most frequently used Hangul hand-written characters among the KS C 2,350 Hangul character set. They were generated by more than 1,000 different writers of different social environments, occupations, and geographical distributions. Each character, written in a 9×9 mm box, has been scanned with a resolution of 300 DPI. Samples of the KU-1 database can be seen at the site: *http://image.korea.ac.kr/database/KU-1*.

4.2. Preliminary experiment

In order to examine the effect of the proposed evaluation method, we have chosen some character samples and let them evaluated by human subjects.

In the preliminary experiments of subjective evaluation, human subjects were presented about 500 samples for each character class to give their evaluation : G1) very good, G2) good, G3) normal(+), G4) normal(-), G5) poor, or G6) very poor. Then, we took the average of the value overall subjects.

Figure 3 shows the two evaluation results which were carried out by human subjects and the proposed evaluation method, respectively. In the figure most samples are classified to the same group by both methods, so we can find out that the proposed evaluation method achieves similar results to the human subjective evaluation.

4.3. Experimental results

In order to verify the performance of the proposed method, the test set has been divided into six data groups,

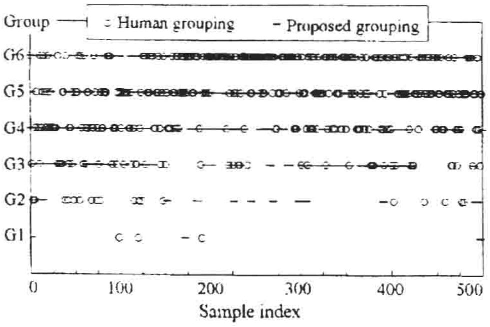


Figure 3. Human subjective evaluation vs. proposed objective evaluation

and then variation have been analyzed. Figure 4 shows the results for each data group.

The results show that the variation increases at linearly from the first group, $G1$ and that the proposed method classifies samples into several groups properly and the measurement of quality is done correctly, using the *extended average entropy(EAE)* in gray-scale. However, the nearly constant rate of variation for each Hangul type is a little different and depends on the complexity of character structure. This means that a character with a complex structure has more variants in shapes than a character with a simple structure. In other words, a complex character has a large variety of strokes, whereas a simple character has a small variety of strokes. From the above observations, we can conclude that the structure of a character is closely related to the variations in its shape.

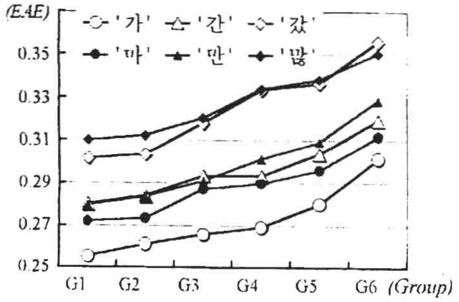


Figure 4. The EAE of each quality group

Figure 5 shows the first 20 samples of each data group organized through the handwriting quality measurement. From this figure, we can verify that the connectivity among strokes and the variety of strokes are more conspicuous, and the separation of phonemes is weaker, as the data differ more from the first group. Thus, it is evident that the proposed method measures the handwriting qualities in accordance with the human criteria.

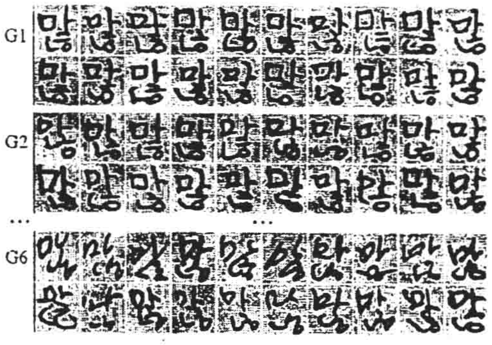


Figure 5. First 20 samples of each data group

5. Concluding remarks

In this paper, we proposed an automatic quality measurement method of gray-scale handwriting data, to offer a means by which different databases can be compared objectively. First, we defined an *extended average entropy(EAE)*, an extension of the *average entropy(AE)* in binary-scale, to directly measure the handwriting qualities in a given gray-scale character database. Second, we measured the quality of each sample in a class, and classified all data within a class into several groups according to their handwriting qualities.

The experimental results confirmed that the proposed method was useful for measuring the qualities of handwritten Hangul characters objectively and automatically. We also showed that the proposed method evaluated the handwriting qualities in accordance with the human criteria, through the preliminary experiment.

References

[1] S. L. Chou and S. S. Yu. Sorting qualities of handwritten Chinese characters for setting up a research database. *Proc. of 2nd ICDAR*, Tsukuba, Japan, 1993, pp. 474–477.

[2] H. Hase, M. Yoneda, and M. Sakai. Evaluation of handprinting variation of characters using variation entropy. *IEICE Transactions D*, J71-D(6):1048–1056, 1988.

[3] T. Kato. Evaluation system for hand-written characters. *Proc. of SPIE/IS&T Conf. on Machine Vision Application in Character Recognition and Industrial Inspection*, San Jose, 1992, pp. 73–82.

[4] D.-H. Kim, E.-J. Kim, and S.-Y. Bang. A variation measure for handwritten character image data using entropy difference. *Pattern Recognition*, 30(1):19–29, 1997.

[5] D.-I. Kim, S.-Y. Kim, and S.-W. Lee. Design and construction of a large-set off-line handwritten Hangul character image database ku-1. *Proc. of National Conf. on Korean Language Information Processing*, Pusan, Korea 1997, pp. 152–159. (in Korean).

Character Pattern Extraction Based on Local Multilevel Thresholding and Region Growing

Hideaki Goto
Education Center for Information Processing,
Tohoku University,
Kawauchi, Aoba, Sendai-shi 980–8576, Japan
hgot@ecip.tohoku.ac.jp

Hirotomo Aso
Graduate School of Engineering,
Tohoku University,
Aramaki, Aoba, Sendai-shi 980–8578, Japan
aso@ecei.tohoku.ac.jp

Abstract

Recent remarkable progress in computer systems and printing devices makes it easier to produce printed documents with various designs. Text characters are often printed on colored backgrounds, and sometimes on complex backgrounds. Some methods have been developed for character extraction from document images and scene images with complex backgrounds. However, those methods are designed to extract rather large characters, and often fails to extract small characters. This paper proposes a new method by which character patterns can be extracted from document images with complex background. The method is based on the local multilevel thresholding and pixel labeling, and the region growing. This framework is very useful for extracting character patterns from badly illuminated document images. The performance of extracting small character patterns has also been improved by suppressing the influence of mixed-color pixels around character edges.

1. Introduction

With the recent remarkable progress in computer systems and printing devices, we have a number of documents in which text characters are printed on colored backgrounds, and on complex background images. Systems for document analysis and recognition require a process which separates text characters from such colored and/or complex backgrounds so that the optical character recognition stage can recognize the texts. Even if a document has a white, uniform background and consists of black texts, the document image inputted by a image scanner or a camera often has uneven brightness due to the uneven illumination. In order to design robust document recognition systems, the character extraction process have to be tolerant of such an uneven brightness of the document image as well.

The intensity gradient based thresholding is very useful for badly illuminated document images [5], however, it cannot segment text characters in the documents with complex backgrounds. Some methods have been developed for the character extraction from color document images. The methods based on isochromatic line analysis cannot handle small characters, because the strokes of small characters are often unclear. The methods based on the color clustering can handle smaller characters, however, they still fail to extract very small characters [3]. The robustness of the process for badly illuminated documents has not been well considered yet.

This paper proposes a new method by which small character patterns can be extracted from document images with complex background. The method is based on the local multilevel thresholding and pixel labeling, and the region growing. This framework is very useful for extracting character patterns from badly illuminated document images.

2. Character pattern extraction based on local multilevel thresholding and region growing

2.1. Detection of local representative gray levels and pixel labeling

The character pattern extraction method proposed in this paper consists of the following four major stages.

1. Local multilevel thresholding and initial pixel labeling.
2. Edge compensation in labeled images.

0-7695-0750-6/00 $10.00 © 2000 IEEE

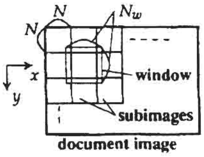


Figure 1. Partitioning of the input image

3. Region growing based on label merging between neighboring subimages.
4. Creation of decomposed images.

At the beginning, the input document image is partitioned into square subimages as shown in Figure 1. Let $P(x,y)$ denote the pixel at the coordinate (x,y), and let $f(x,y)$ denote the intensity of the pixel. Let R^i denote the i-th subimage. The input image is assumed to be a 400dpi, 8bit grayscale image. We used $N = 50$ (3.2mm on the document) as the subimage size, and $N_w = 60$ as the window size, since they seemed to be appropriate in our preliminary experiments.

The weighted histogram of the pixel intensity is calculated in every window co-centered by each subimage. For the pixel at (x,y), the weight is defined as $w(x,y) = \max(1 - \alpha E(x,y), 0)$, where $E(x,y)$ denotes the intensity component of the Sobel's edge detection operator, and α denotes a positive coefficient. We had verified that the process is not so sensitive to α, and used $\alpha = 1/32$. The weighted histogram is expected to be useful for the extraction of small characters. The reason is as described below.

The character patterns in the document image are slightly or strongly blurred in general. There are a lot of mixed-color pixels whose gray level (or color) is the mixture of the background's and the character's. Due to the mixed-color pixels, the normal histogram has some spurious peaks between the character's gray level (about 60) and the background's gray level (about 230) as shown by the broken line in Figure 2. If we apply the multilevel thresholding using the peaks in the histogram, the spurious peaks will split thin strokes of characters into several gray levels, and make it difficult to extract the character strokes correctly. If the document image is blurred more, the valley between the two gray levels will be filled, and consequently, it will be impossible to discriminate the characters and the background. The weighted histogram will help making the valley clear, because it suppresses the counts of the mixed-color edge pixels (Figure 2).

The calculated histogram is smoothed by the moving average and the hysteresis smoothing. The width of the moving average is 7. The half width of the cursor in hysteresis smoothing is 20% of the height of the histogram. Then, the peaks in smoothed histogram are found and the peak positions are regarded as the representative gray levels. Let D_j^i denote the gray level of the j-th peak in the i-th subimage. For each pixel $P(x,y)$, the number j which minimize $|f(x,y) - D_j^i|$ is found, and the pixel is tagged with the label (i,j). The gray level of the pixel is turned into $f'(x,y) = D_j^i$.

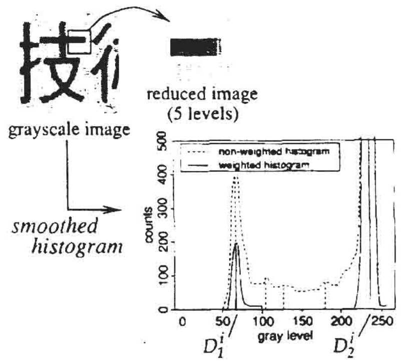


Figure 2. Detection of representative gray levels at each subimage

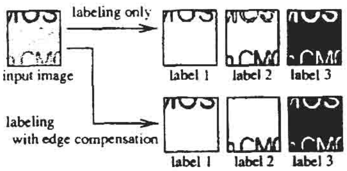


Figure 3. The effect of edge compensation

2.2. Edge compensation in labeled images

If more than one foreground image coexist in the same subimage and if their gray levels differ, the result of the multilevel thresholding often shows undesirable split of the foreground images. In Figure 3, note that the character stokes above are split into two labels; the skeletons and the edges. We cannot avoid such a result only tuning up the multilevel thresholding, because the spatial information is not taken into account in the thresholding.

In order to reduce the splits of the strokes, the edge compensation process is applied to the initially labeled subimages. Each labeled subimage is scanned in horizontal and vertical ways, and the edge compensation is performed on every scan line (Figure 4). The pixels on the edge are relabeled using two gray levels adjacent to the edge. The "edge" here is such that the length is less than $T_{sp} = 7$ and the length of every step is less than $T_{st} = 3$. The splits of the strokes will be suppressed by the edge compensation as shown in Figure 3.