Optical character recognition with feature extraction and associative memory matrix

Osami Sasaki, MEMBER SPIE
Akihito Shibahara
Takamasa Suzuki, MEMBER SPIE
Niigata University
Faculty of Engineering
8050 Ikarashi 2
Niigata-shi 950-21
Japan
E-mail: osami@eng.niigata-u.ac.jp

Abstract. A method is proposed in which handwritten characters are recognized using feature extraction and an associative memory matrix. In feature extraction, simple processes such as shifting and superimposing patterns are executed. A memory matrix is generated with singular value decomposition and by modifying small singular values. The method is optically implemented with two liquid crystal displays. Experimental results for the recognition of 26 handwritten alphabet characters clearly show the effectiveness of the method. © *1998 Society of Photo-Optical Instrumentation Engineers.* [S0091-3286(98)00206-2]

Subject terms: pattern recognition; feature extraction; associative memory; optical signal processing; singular value decomposition.

Paper 23097 received Sep. 19, 1997; revised manuscript received Dec. 29, 1997; accepted for publication Dec. 29, 1997.

1 Introduction

Matched-filter-based correlators, which have been widely used for optical pattern recognition, require complex filter fabrication and as many filters as the number of reference patterns. To alleviate the requirement of complex filter fabrication, the joint transform correlator has been devised and is effective for real-time processing. To reduce the number of filters, feature-extracted patterns are used as reference patterns instead of the target pattern itself, and correlation-based methods are used to obtain the correlation between input patterns and feature-extracted patterns.[1-3] Although correlation-based methods provide correlation invariance under transformation of input patterns, they are highly sensitive to pattern deformation. Because of this property, different methods are necessary for the recognition of handwritten characters.

In this paper, we propose a method for the recognition of handwritten characters in which feature extraction and an associative memory matrix are used. Since information concerning the position and magnitude of line components included in input patterns are used as their features, the proposed method is somewhat insensitive to the deformation of patterns arising in handwritten characters. In addition, a method to extract the features is much simpler than correlation-based methods, where multiplexed matched spatial filtering[1] or multichannel joint transform correlator[2] are required. We extract the features using very simple processes such as shifting and dividing input patterns and computing a product of two binary patterns. After the feature extraction process, we obtain a feature vector that is related to line components of an input pattern. Although this feature vector is not greatly influenced by deformations of the input patterns, it can not clearly identify input characters. A feature vector becomes an input to an associative memory matrix so that its output vector identifies the input character. A memory matrix is generated with singular value decomposition[4] and by modifying small singular values[5] to produce a memory matrix insensitive to noise in feature vectors due to input pattern deformation. The method for the recognition of handwritten characters is easily implemented with an optical system using two liquid crystal displays (LCDs) to compute the product of two patterns and a computer to execute simple pattern processing.

2 Input Pattern Feature Extraction

A handwritten capital alphabet character is detected with a CCD camera and its image is transformed to a binary pattern of 10-× 10-pixel size, as shown in Figs. 1(a) and 1(b). This binary pattern is termed the input pattern. To obtain a line pattern representing a vertical line component, we shift the input pattern to the upper side by 1 pixel, as shown in Fig. 1(c). This shifted pattern is superimposed on the input pattern. If the two pixels of the input pattern and the shifted pattern at the same position are both 1, then the pixel of a line pattern at the position is 1. If the two pixels are in other conditions, the pixel of a line pattern is 0. The resulting line pattern is obtained as shown in Fig. 1(d). There are three other directions for shifting the input pattern by 1 pixel, i.e., right, right upper corner, and left upper corner. From these shifts, we obtain three line patterns pertaining to a horizontal line, a line along right upper corner, and a line along left upper corner.

To extract information concerning the position and magnitude of a vertical line from a line pattern related to that vertical line, we use three kinds of position patterns P_i ($i=1$ to 3), as shown in Fig. 2. With the position pattern P_1 superimposed on the line pattern of Fig. 1(d), we detect how many pixels are 1 at an identical position. In this example, this number, given by R_1, is 4, and it is referred to as the corresponding feature value. We have three kinds of position patterns P_i ($i=4$ to 6) for a horizontal line and four kinds of position patterns P_i ($i=7$ to 11) and P_i ($i=11$ to 14) for lines inclined at 45 and 135

 © 1998 Society of Photo-Optical Instrumentation Engineers

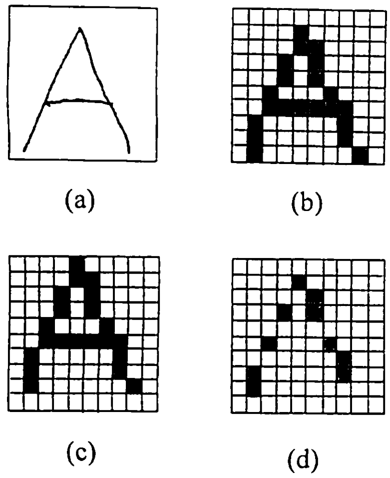

Fig. 1 Process to obtain a line pattern representing the vertical line component: (a) handwritten character, (b) input pattern, (c) shifted pattern, and (d) line pattern.

deg, respectively, as shown in Fig. 2. For the input pattern, we have position pattern P_{15}, which detects how many pixels of 1 the input patterns contain in the specified region. Thus, we obtain feature values R_i ($i=1$ to 15) for an input pattern. Next we form a feature vector $\mathbf{x}$ whose element are given by

$$x_i = R_i \ (i=1 \text{ to } 6), \quad x_i = 2R_i \ (i=7 \text{ to } 14), \quad x_i = R_i/2 \ (i=15). \tag{1}$$

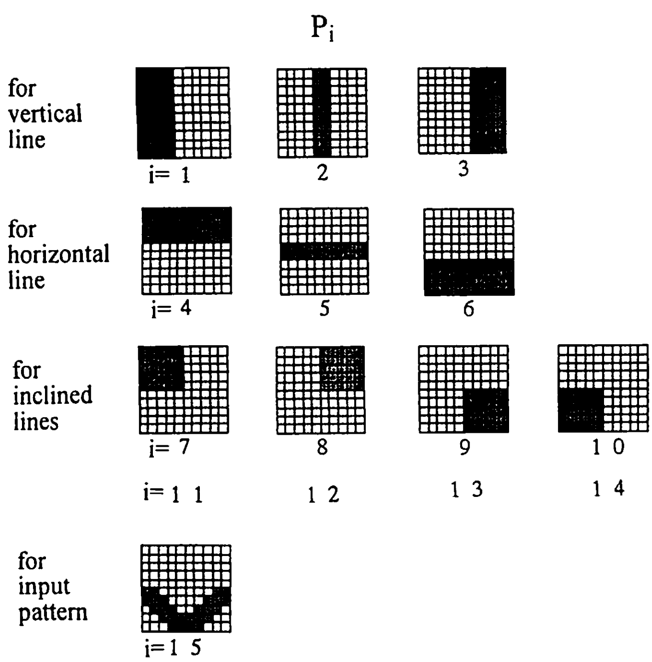

Fig. 2 Position patterns P_i ($i=1$ to 3) for a vertical line, P_i ($i=4$ to 6) for a horizontal line, P_i ($i=7$ to 10) for a line along the right upper corner, P_i ($i=11$ to 14) for a line along the left upper corner, and P_{15} for the input pattern.

We divide 26 alphabet characters into two groups based on the line components contained in the characters. For this grouping, we consider the value

$$G = \sum_{i=1}^{6} R_i \Big/ \sum_{i=7}^{14} R_i. \tag{2}$$

In this paper, group II consists of eight characters of E, F, H, I, L, P, T, and U, which contain less inclined lines. Handwritten characters of group II have a high possibility that the value of G is large. Group I consists of the others of 26 alphabet characters and P. After we obtain the feature values, the group to which the input pattern belongs is decided by investigating whether the value of G is larger than a specified value γ. Computer simulations determine an appropriate value of γ.

3 Recognition with an Associative Memory Matrix

After judging whether an input character belongs to group I or group II, an input character is recognized by producing an output vector $\mathbf{y}_i$ from a feature vector $\mathbf{x}$ with an associative memory matrix $\mathbf{M}$ as

$$\mathbf{y} = \mathbf{M}\mathbf{x}. \tag{3}$$

A memory matrix is generated with singular value decomposition (SVD). When K feature vectors $\mathbf{x}_i$ corresponding to K output vectors $\mathbf{y}_i$ ($i=1$ to K), respectively, are memorized in memory matrix $\mathbf{M}$, we have the relation

$$\mathbf{Y} = \mathbf{M}\mathbf{X}, \tag{4}$$

where

$$\mathbf{X} = [\mathbf{x}_1 \ldots \mathbf{x}_K], \quad \mathbf{Y} = [\mathbf{y}_1 \ldots \mathbf{y}_K]. \tag{5}$$

The i'th element of the output vector $\mathbf{y}_i$ is 1, and the others are 0, which shows the corresponding input character is the i'th one of K characters. Then the output matrix $\mathbf{Y}$ becomes a unit matrix. If a pseudo inverse $\mathbf{X}^+$ of the input matrix is obtained with SVD, a memory matrix is given by

$$\mathbf{M} = \sum_{i=1}^{K} c_i \mathbf{U}_i, \tag{6}$$

where

$$c_1 \leqslant c_2 \leqslant \ldots \leqslant c_K. \tag{7}$$

When input characters are different from memorized characters, input characters contain noise. This noise component is amplified by the coefficients c_i, whose values are large. To eliminate effects of noise on output vectors, J coefficients whose values are larger than the others are replaced with a nonzero value of α to obtain a memory matrix

$$\mathbf{M} = \sum_{i=1}^{K-J} c_i \mathbf{U}_i + \sum_{i=K-J+1}^{K} \alpha \mathbf{U}_i. \tag{8}$$

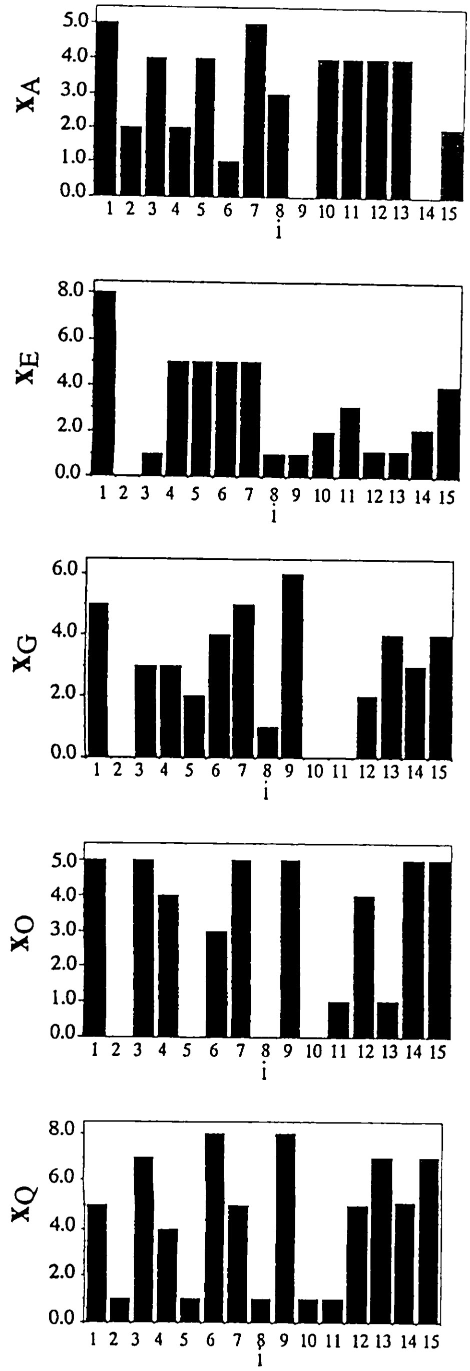

Fig. 3 Memorized feature vector $\mathbf{x}_i$ for characters of i=A, E, G, O, and Q.

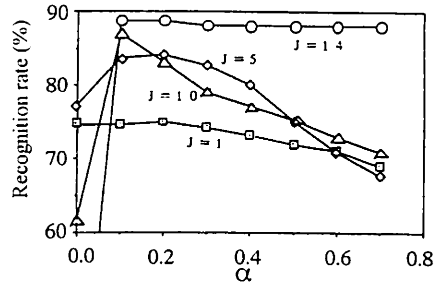

Fig. 5 Mean recognition rate versus value of α for different values of J.

In this paper, a memory matrix $\mathbf{M}_{\mathrm{I}}$ for group I is obtained with Eq. (8). A memory matrix $\mathbf{M}_{\mathrm{II}}$ for group II is obtained with Eq. (6), since all coefficients are small. Details are described in the next section.

4 Memory Matrix Determination through Computer Simulations

First, we prepared 12 handwritten characters for one alphabet character, and we obtained 12 feature vectors using the process described in Sec. 2. Averaging 12 feature vectors provided a feature vector which is memorized in memory matrix. Memorized feature vector $\mathbf{x}_i$ for characters of i = A, E, G, O, and Q are shown in Fig. 3. Position pattern P_{15} is useful to detect the existence of a semicircular line in lower part of input patterns. Classifying into the two groups is carried out for $\gamma = 0.4$.

Next, we determined memory matrices $\mathbf{M}_{\mathrm{I}}$ and $\mathbf{M}_{\mathrm{II}}$ for groups I and II, respectively. For group II, coefficients c_i ($i = 1$ to 8) of Eq. (6) were all almost equal and small. Thus, a memory matrix $\mathbf{M}_{\mathrm{II}}$ was generated using Eq. (6). For group I, there are 19 coefficients, and values of c_i ($i = 1$ to 14) are shown in Fig. 4. Since values of c_i ($i \geq 15$) were very large, we used Eq. (8) to generate the memory matrix $\mathbf{M}_{\mathrm{II}}$. We tried to recognize 15 handwritten characters for one alphabet character, and calculated the ratio of the number of correctly recognized characters to the total number of characters as the recognition rate. Mean recognition rate over 26 alphabet characters was used to estimate the ability of memory matrix. Figure 5 shows the mean recognition rate versus the value of α for different values of J. This result indicates that $\alpha = 0.1$ is the best value. Figure 6 shows mean recognition rate versus J at

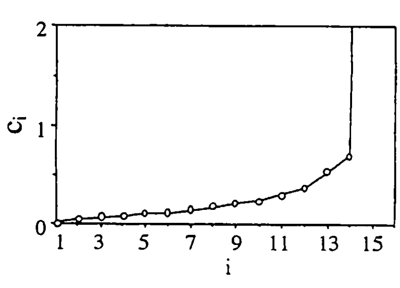

Fig. 4 Values of coefficients c_i ($i = 1$ to 14) for group I.

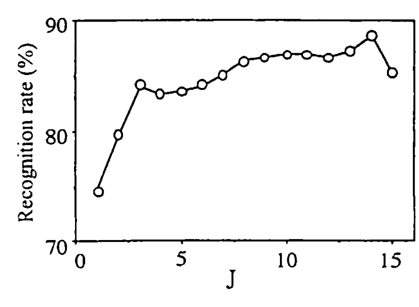

Fig. 6 Mean recognition rate versus J at $\alpha = 0.1$.

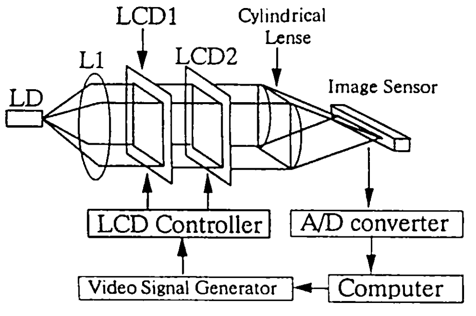

Fig. 7 Optical character recognition system.

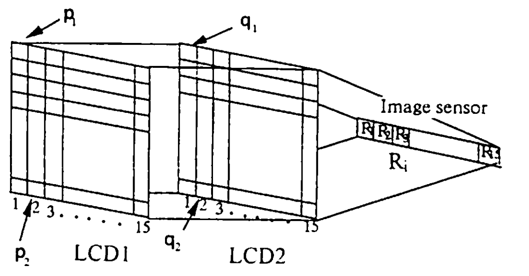

Fig. 9 Optical computation to obtain feature values R_i from vectors $\mathbf{p}_i$ and $\mathbf{q}_i$ with two LCDs.

$\alpha=0.1$, which indicates that $J=14$ is the best value. Therefore, we generated memory matrix $\mathbf{M}_{\mathrm{I}}$ for group I from Eq. (8) at $\alpha=0.1$ and $J=14$.

5 Optical Implementation of the Method with LCDs

We constructed the optical system shown in Fig. 7, which performs feature extraction of an input pattern and the necessary matrix-vector computations. Binary patterns and memory matrices to be displayed on the LCDs are formed in a personal computer. Light from a laser diode (LD) are collimated with lens 1 ($L1$). The collimated light passes through the two LCDs and the transmitted light is collected with a cylindrical lens onto the image sensor. Then the output of image sensor is fed to the personal computer through an analog-to-digital (A/D) converter.

First, we explain how to obtain feature vectors with Figs. 8 and 9. In Fig. 8 the input pattern is a character of E, and a shifted pattern is formed by shifting the input pattern upward by one pixel element. A part of the input pattern, which is specified by the position pattern P_1, is taken out, and we form a column vector $\mathbf{p}_1$ by scanning the pixels of the pattern. In the same way, we form a column vector $\mathbf{q}_1$, which represents a pattern taken out from the shifted pattern according to position pattern P_1. For position pattern P_2, we have column vectors $\mathbf{p}_2$ and $\mathbf{q}_2$. Thus, for all position patterns except $\mathbf{P}_{15}$, we have column vectors $\mathbf{p}_i$ and $\mathbf{q}_i$ ($i=1$ to 14). For position pattern P_{15}, $\mathbf{q}_{15}$ is equal to $\mathbf{p}_{15}$, which represents a part of the input pattern specified by P_{15}. These simple input pattern manipulations are carried out with a personal computer. Vector $\mathbf{p}_i$ is displayed in a region of LCD1, denoted with number i in the vertical direction, as shown in Fig. 9. Vector $\mathbf{q}_i$ is also displayed in the same region of LCD2. By transmitting a collimated light through the two LCDs, we obtain the intensity distribution on the image sensor, which shows feature values R_i ($i=1$ to 15). From the output of the image sensor we obtain a feature vector $\mathbf{x}$ of Eq. (1) with a computer.

Next, we explain how to perform the matrix-vector computation given by Eq. (3). The elements of matrix $\mathbf{M}$ are divided into nonnegative ones and negative ones. The matrix is decomposed as $\mathbf{M}=\mathbf{M}_{(+)}-\mathbf{M}_{(-)}$. Here $\mathbf{M}_{(+)}$ consists of nonnegative elements of $\mathbf{M}$ where the negative elements of $\mathbf{M}$ are replaced by 0's. On the other hand, $\mathbf{M}_{(-)}$ consists of negative elements, with signs changed and the nonnegative elements of $\mathbf{M}$ are replaced by 0's. An input character $\mathbf{x}$ is displayed on LCD1 in the region denoted by the notations x_1 to x_n ($n=15$) in the vertical direction, and the same input character $\mathbf{x}$ is repeated m times in the horizontal direction, as shown in Fig. 10. When we use matrices $\mathbf{M}_{\mathrm{I}}$ and $\mathbf{M}_{\mathrm{II}}$ as memory matrix $\mathbf{M}$, $m=19$ and 8, respectively. Output of the image sensor is $\mathbf{y}_{(+)}$ and $\mathbf{y}_{(-)}$ when $\mathbf{M}_{(+)}$ and $\mathbf{M}_{(-)}$ respectively, are displayed on LCD2 with 256 gray levels. Finally, output $y_i=y_{i(+)}-y_{i(-)}$ ($i=1$ to 26) is computed. If the element y_i corresponding to the input pattern has the largest value among the other elements of the final output, recognition is accomplished.

6 Experimental Results

The processing carried out in the system shown in Fig. 7 is as follows: (1) read a handwritten character; (2) form an input pattern; (3) form vectors $\mathbf{p}_i$ and $\mathbf{q}_i$ ($i=1$ to 15) and display them; (4) detect feature values R_i ($i=1$ to 15); (5)

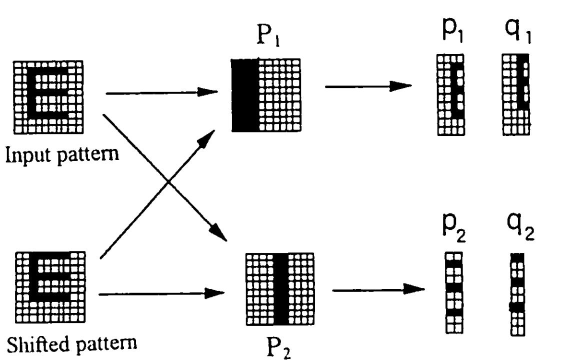

Fig. 8 Generation of vectors $\mathbf{p}_i$ and $\mathbf{q}_i$ for extracting line component features.

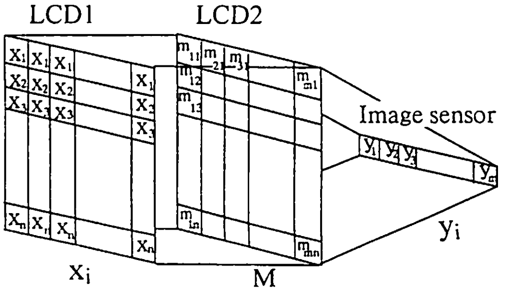

Fig. 10 Optical matrix-vector computation to obtain output vector $\mathbf{y}_i$ from the feature vector and the memory matrix with two LCDs.

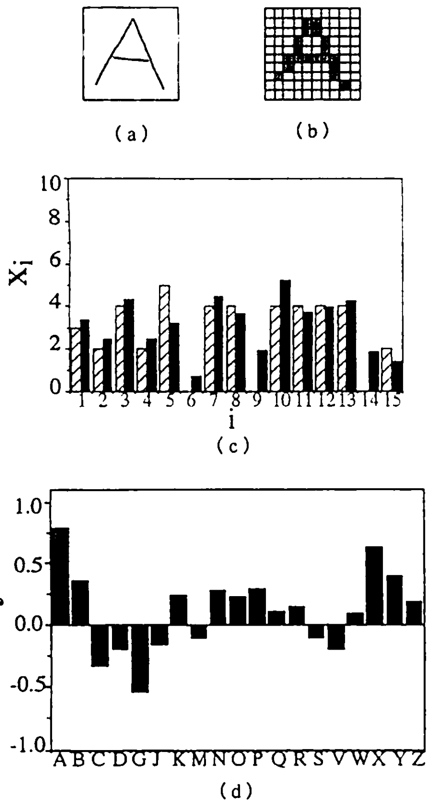

Fig. 11 Result I for recognition of the character A.

form a feature vector and display it; (6) calculate G, decide the group, and display memory matrix $\mathbf{M}_{(+)}$; (7) detect output $\mathbf{y}_{(+)}$; (8) display memory matrix $\mathbf{M}_{(-)}$; (9) detect output $\mathbf{y}_{(-)}$; and (10) calculate final output $y_i = y_{i(+)} - y_{i(-)}$ ($i = 1$ to 26). Experimental results are shown in Figs. 11–16. Figures 11(a) and 11(b) show the handwritten character and the input pattern, respectively. Feature vectors obtained from the input pattern with the optical system and computer simulation are shown in Fig. 11(c) by solid and striped bars, respectively. By comparing these two results we can see the accuracy of the optical computation. There are some errors in the optical computation, but these errors do not affect ultimate results of recognition. The value of G was larger than 0.4, so the input pattern belonged to group I. Figure 11(d) shows a final output $\mathbf{y}$. The horizontal axis represents i'th element of $\mathbf{y}$, and instead of denoting the number of i, the corresponding characters belonging to group I are denoted. A maximum value appears in the position of the character A, so we recognized the input pattern as A. Results pertaining to another variation of the character A is shown in Fig. 12. We can recognize the characters from the feature vectors, which are a little different from the feature vector memorized in the memory matrix shown in Fig. 3(a). A characteristic point common to all of the feature vectors of the character A is that the values of x_6 are very small. Results for recognition of two variations of the character E are shown in Figs. 13 and 14. For these feature values, G was less than 0.4, so these input patterns were in group II. Characters belonging to group II are labeled along the horizontal axis of Figs. 13(d) and 14(d). Similarly obtained results for recognition of the

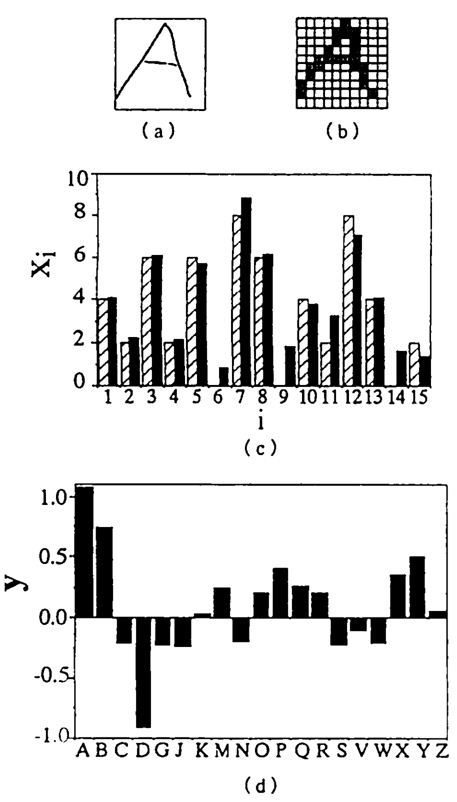

Fig. 12 Result II for recognition of the character A.

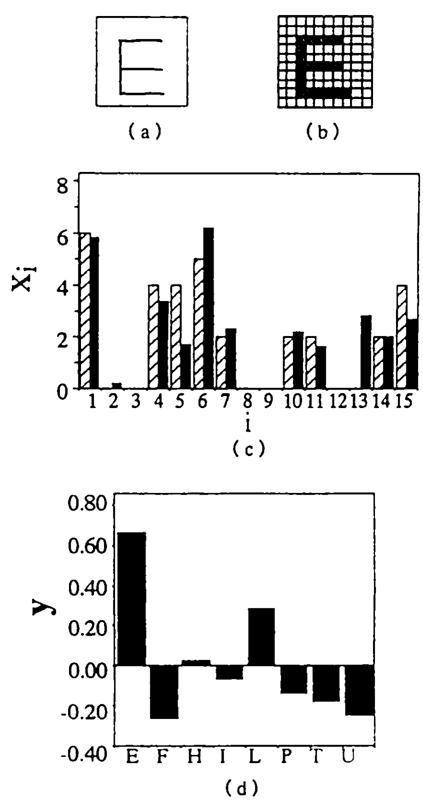

Fig. 13 Result I for recognition of the character E.

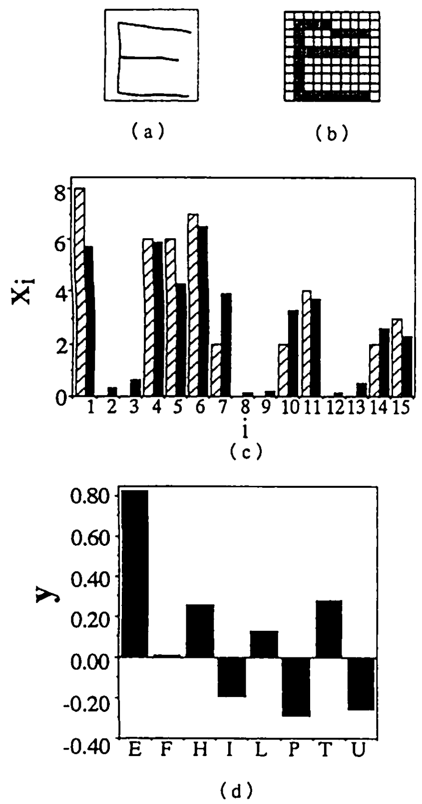

Fig. 14 Result II for recognition of the character E.

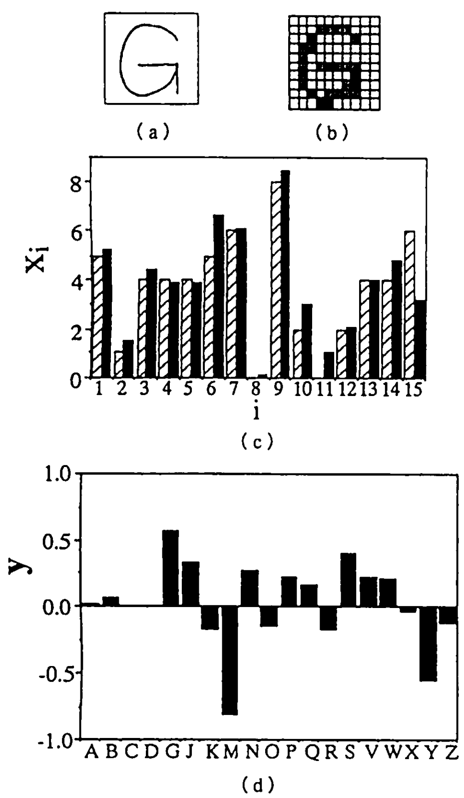

Fig. 15 Result I for recognition of the character G.

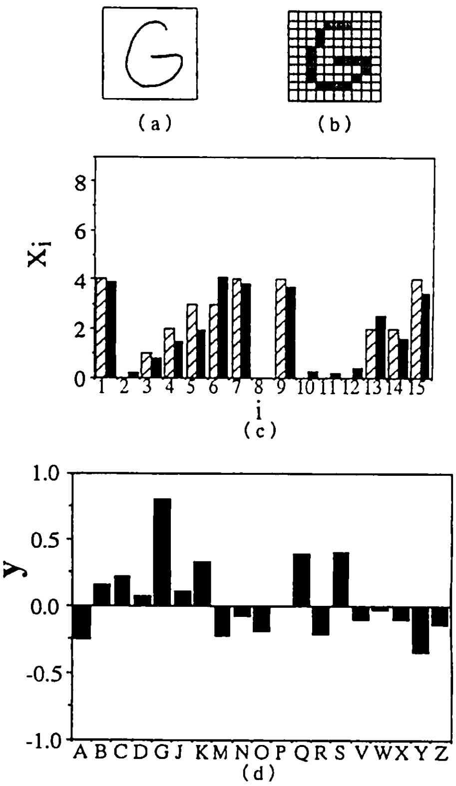

Fig. 16 Result II for recognition of the character G.

character G are shown in Figs. 15 and 16. All of the results indicate that the method proposed here can recognize handwritten characters. We tried to recognize 15 variations of one alphabet character to obtain a recognition rate for one alphabet character. Figure 17 shows the recognition rate for all 26 characters. The recognition rate averaged over 26 alphabet characters was 87%. The recognition rate for the characters O and R is low, in part because the characters O and R are similar to Q and P, respectively. In the experiments, the pixel size of the input patterns was 10×10. The recognition rate can be improved by increasing the input pattern pixel size, which leads to extracting more character fine features into the feature vector values. Computer simulations showed that the recognition rate for the characters O and R was improved to about 75% and the recognition rate averaged over 26 alphabet characters was 92% when the pixel size was 20×20.

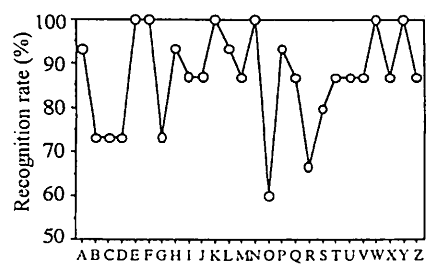

Fig. 17 Recognition rate for 26 alphabet characters.

7 Conclusion

We proposed a method to recognize handwritten characters using feature extraction and an associative memory matrix. Since the features used were information concerning the position and magnitude of line components included in the input patterns, the processing required to obtain the feature vectors was very simple compared to correlation-based methods. The simple processing was optically implemented with two LCDs. Appropriate memory matrices were generated from SVD through computer simulations. We tried 26 handwritten alphabet characters, and the results clearly showed the effectiveness of the method. Increasing the input pattern pixel sizes would produce higher recognition rates in experiments.

References

1. S. Kamemaru, H. Itoh, and J. Yano, "Character recognition by feature extraction using cross-correlation signals from a matched filters," *Opt. Eng.* **32**, 26–32 (1993).
2. M. S. Alam, "Feature-extracted joint transform correlation," *Appl. Opt.* **34**, 8148–8153 (1995).
3. K. M. Iftekharuddin, T. D. Schechinger, K. Jemili, and M. A. Karim, "Feature-based neural wavelet optical character recognition system," *Opt. Eng.* **34**, 3193–3198 (1995).
4. T. Kohonen and M. Ruohonen, "Representation of associated pairs by matrix operators," *IEEE Trans. Comput.* **C-22**, 701–702 (1973).
5. C. K. Rushforth, A. E. Crawford, and Y. Zhou, "Least-squares reconstruction of objects with missing high-frequency components," *J. Opt. Soc. Am.* **72**, 204–211 (1982).

Osami Sasaki received his BE and ME degrees in electrical engineering from the Niigata University in 1972 and 1974, respectively, and his DrEng degree in electrical engineering from the Tokyo Institute of Technology in 1981. He is a professor of electrical and electronic engineering at the Niigata University. His research interests include optical metrology, and optical information processing.

Akihito Shibahara received his BE and ME degrees in electric engineering from Niigata University in Japan in 1975 and 1993, respectively. His research involved optical pattern recognition. Since 1975 he has worked on the development of RISC system software and tools at Semiconductor System Engineering Center, Toshiba Corporation.

Takamasa Suzuki received his BE degree in electrical engineering from the Niigata University in 1982, his ME-degree in electrical engineering from the Tohoku University in 1984, and his DrEng degree in electrical engineering from the Tokyo Institute of Technology in 1994. He is an associate professor of electrical and electronic engineering at the Niigata University. His research interests include optical metrology, optical information processing, and phase conjugate optics.