PAPER

Score-Level Fusion of Phase-Based and Feature-Based Fingerprint Matching Algorithms

Koichi ITO[†a)], ***Member***, **Ayumi MORITA**[††], ***Nonmember***, **Takafumi AOKI**[†], ***Member***, **Hiroshi NAKAJIMA**[††], ***Nonmember***, **Koji KOBAYASHI**[††], ***Member***, ***and*** **Tatsuo HIGUCHI**[†††], ***Fellow***

SUMMARY This paper proposes an efficient fingerprint recognition algorithm combining phase-based image matching and feature-based matching. In our previous work, we have already proposed an efficient fingerprint recognition algorithm using Phase-Only Correlation (POC), and developed commercial fingerprint verification units for access control applications. The use of Fourier phase information of fingerprint images makes it possible to achieve robust recognition for weakly impressed, low-quality fingerprint images. This paper presents an idea of improving the performance of POC-based fingerprint matching by combining it with feature-based matching, where feature-based matching is introduced in order to improve recognition efficiency for images with nonlinear distortion. Experimental evaluation using two different types of fingerprint image databases demonstrates efficient recognition performance of the combination of the POC-based algorithm and the feature-based algorithm.

key words: *fingerprint recognition, phase-only correlation, feature-based matching, combination of matchers, score-level fusion, biometrics*

1. Introduction

Biometric authentication has been receiving extensive attention over the past decade with increasing demands in automated personal identification. Biometrics is to identify individuals using physiological or behavioral characteristics, such as fingerprint, face, iris, retina, palmprint, gait, voice etc. Among all the biometric techniques, fingerprint recognition [1], [2] is the most popular method and is successfully used in many applications.

Major approaches for fingerprint recognition today can be broadly classified into feature-based approach [3], [4] and correlation-based approach [5], [6]. Typical fingerprint recognition methods employ feature-based matching, where minutiae (i.e., ridge ending and ridge bifurcation) are extracted from the registered fingerprint image and the input fingerprint image, and the number of corresponding minutia pairs between the two images is used to recognize a valid fingerprint image [1]. The feature-based matching is highly robust against nonlinear fingerprint distortion, but shows only limited capability for recognizing poor-quality fingerprint images with low S/N due to unexpected fingertip conditions (e.g., dry fingertips, rough fingertips, allergic-skin fingertips) as well as weak impression of fingerprints. On the other hand, as one of the efficient correlation-based approaches [5], [6], we have proposed a fingerprint recognition algorithm using Phase-Only Correlation (POC) [6]–[9] — an image matching technique using the phase components in 2D Discrete Fourier Transforms (2D DFTs) of given images —, and developed commercial fingerprint verification units for access control applications [10]. Historically, the POC-based image matching has been successfully applied to high-accuracy image registration tasks for computer vision applications [11]–[13]. The use of Fourier phase information of fingerprint images makes possible highly reliable fingerprint matching for low-quality fingerprints whose minutiae are difficult to be extracted as mentioned above. However, the performance of the POC-based fingerprint matching is degraded by nonlinear distortion in fingerprint images.

Each approach employs different matching criteria to compute a matching score which is used for authentication, since the minutiae-based matching uses local information of a fingerprint while the correlation-based matching uses global information of a fingerprint. In pattern recognition literature, it is often observed that different classifiers with the same performance misclassify different patterns [14]. This implies that different classifiers provide complementary information about the classification task. Hence, a combination approach employing various information of pattern could improve the overall system performance. According to this idea, there are some papers discussing the algorithm combining multiple fingerprint matchers [15]–[18]. In the papers [16], [17], minutia information and ridge feature map are used to improve the overall performance of fingerprint matching system. The other related paper [18] describes the analytical study of combining multiple fingerprint matching algorithms submitted to FVC (Fingerprint Verification Competition) 2004 [19]. All these works suggest that the best performance may be obtained by combining minutiae-based and correlation-based approaches. However, no experimental evaluation of the combination of minutiae-based and correlation-based approaches has been provided. On the other hand, this paper presents a novel fingerprint recognition algorithm combining POC-based image matching and feature-based matching to improve matching performance for both fingerprint images with poor image quality and

Manuscript received July 6, 2009.
Manuscript revised October 31, 2009.
[†]The authors are with the Department of Computer and Mathematical Sciences, Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980–8579 Japan.
[††]The authors are with Yamatake Corporation, Fujisawa-shi, 251–8522 Japan.
[†††]The author is with the Faculty of Engineering, Tohoku Institute of Technology, Sendai-shi, 982-8577 Japan.
a) E-mail: ito@aoki.ecei.tohoku.ac.jp
DOI: 10.1587/transfun.E93.A.607

with nonlinear shape distortion. In this algorithm, two approaches are expected to play a complementary role and result in significant improvement of recognition performance. To combine fingerprint matching algorithms, we employ a score-level fusion method which is one of the most convenient method for combination of matchers [15]. Experimental evaluation using two different types of fingerprint image databases demonstrates efficient recognition performance of the proposed algorithm.

The rest of the paper is organized as follows: Sect. 2 describes a phase-based fingerprint matching algorithm. Section 3 gives a brief description of the three feature-based fingerprint matching algorithms to be combined with the phase-based fingerprint matching algorithm. Section 4 describes a proposed fingerprint matching algorithm combining POC-based image matching and feature-based matching. Section 5 demonstrates a set of experiments for evaluating matching performance of the combined algorithm. Section 6 ends with some concluding remarks.

2. Phase-Based Fingerprint Matching

In this section, we introduce the principle of phase-based image matching using the Phase-Only Correlation (POC) function (which is sometimes called the "phase-correlation function") [11]–[13]. We also describe the POC-based fingerprint matching algorithm.

2.1 Fundamentals of Phase-Only Correlation

Consider two $N_1 \times N_2$ images, $f(n_1, n_2)$ and $g(n_1, n_2)$, where we assume that the index ranges are $n_1 = -M_1, \cdots, M_1$ ($M_1 > 0$) and $n_2 = -M_2, \cdots, M_2$ ($M_2 > 0$), and hence $N_1 = 2M_1 + 1$ and $N_2 = 2M_2 + 1$. Note that we assume here the sign symmetric index ranges $\{-M_1, \cdots, M_1\}$ and $\{-M_2, \cdots, M_2\}$ for mathematical simplicity. The discussion could be easily generalized to non-negative index ranges with power-of-two image size. Let $F(k_1, k_2)$ and $G(k_1, k_2)$ denote the 2D DFTs of the two images. $F(k_1, k_2)$ and $G(k_1, k_2)$ are given by

$$F(k_1, k_2) = \sum_{n_1=-M_1}^{M_1} \sum_{n_2=-M_2}^{M_2} f(n_1, n_2) W_{N_1}^{k_1 n_1} W_{N_2}^{k_2 n_2} = A_F(k_1, k_2) e^{j\theta_F(k_1, k_2)}, \quad (1)$$

$$G(k_1, k_2) = \sum_{n_1=-M_1}^{M_1} \sum_{n_2=-M_2}^{M_2} g(n_1, n_2) W_{N_1}^{k_1 n_1} W_{N_2}^{k_2 n_2} = A_G(k_1, k_2) e^{j\theta_G(k_1, k_2)}, \quad (2)$$

where $k_1 = -M_1, \cdots, M_1$, $k_2 = -M_2, \cdots, M_2$, $W_{N_1} = e^{-j\frac{2\pi}{N_1}}$, and $W_{N_2} = e^{-j\frac{2\pi}{N_2}}$. $A_F(k_1, k_2)$ and $A_G(k_1, k_2)$ are amplitude components, and $\theta_F(k_1, k_2)$ and $\theta_G(k_1, k_2)$ are phase components. The normalized cross-power spectrum $R_{FG}(k_1, k_2)$ between $F(k_1, k_2)$ and $G(k_1, k_2)$ is given by

$$R_{FG}(k_1, k_2) = \frac{F(k_1, k_2)\overline{G(k_1, k_2)}}{|F(k_1, k_2)\overline{G(k_1, k_2)}|} = e^{j\theta(k_1, k_2)}, \quad (3)$$

where $\overline{G(k_1, k_2)}$ denotes the complex conjugate of $G(k_1, k_2)$ and $\theta(k_1, k_2)$ denotes the phase difference $\theta_F(k_1, k_2) - \theta_G(k_1, k_2)$. The POC function $r_{fg}(n_1, n_2)$ is 2D Inverse DFT of $R_{FG}(k_1, k_2)$ and is given by

$$r_{fg}(n_1, n_2) = \frac{1}{N_1 N_2} \sum_{k_1=-M_1}^{M_1} \sum_{k_2=-M_2}^{M_2} R_{FG}(k_1, k_2) \times W_{N_1}^{-k_1 n_1} W_{N_2}^{-k_2 n_2}. \quad (4)$$

When two images are similar, their POC function gives a distinct sharp peak. (When $f(n_1, n_2) = g(n_1, n_2)$, the POC function r_{fg} becomes the Kronecker delta function.) When two images are not similar, the peak drops significantly. The height of the peak can be used as a good similarity measure for image matching, and the location of the peak shows the translational displacement between the two images. Other important properties of POC used for biometric authentication tasks are that the POC-based image matching is not influenced by image shift and brightness change, and it is highly robust against noise. See Ref. [6] for detailed discussions.

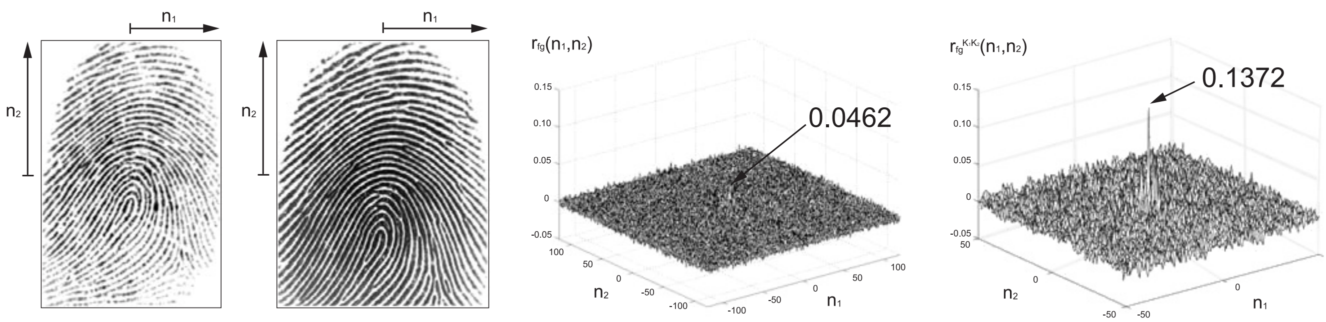
We modify the definition of POC function to have a BLPOC (Band-Limited Phase-Only Correlation) function dedicated to fingerprint matching tasks. The idea to improve the matching performance is to eliminate meaningless high frequency components in the calculation of cross-phase spectrum $R_{FG}(k_1, k_2)$ depending on the inherent frequency components of fingerprint images [6]. Assume that the ranges of the inherent frequency band are given by $k_1 = -K_1, \cdots, K_1$ and $k_2 = -K_2, \cdots, K_2$, where $0 \leq K_1 \leq M_1$ and $0 \leq K_2 \leq M_2$. Thus, the effective size of frequency spectrum is given by $L_1 = 2K_1 + 1$ and $L_2 = 2K_2 + 1$. The BLPOC function is given by

$$r_{fg}^{K_1 K_2}(n_1, n_2) = \frac{1}{L_1 L_2} \sum_{k_1=-K_1}^{K_1} \sum_{k_2=-K_2}^{K_2} R_{FG}(k_1, k_2) W_{L_1}^{-k_1 n_1} W_{L_2}^{-k_2 n_2}, \quad (5)$$

where $n_1 = -K_1, \cdots, K_1$ and $n_2 = -K_2, \cdots, K_2$. Note that the maximum value of the correlation peak of the BLPOC function is always normalized to 1 and does not depend on L_1 and L_2. Also, the translational displacement between the two images can be estimated by the correlation peak position. Figures 1 and 2 show examples of genuine matching and impostor matching using the original POC function r_{fg} and the BLPOC function $r_{fg}^{K_1 K_2}$, respectively. The BLPOC function provides the higher correlation peak and better discrimination capability than the original POC function.

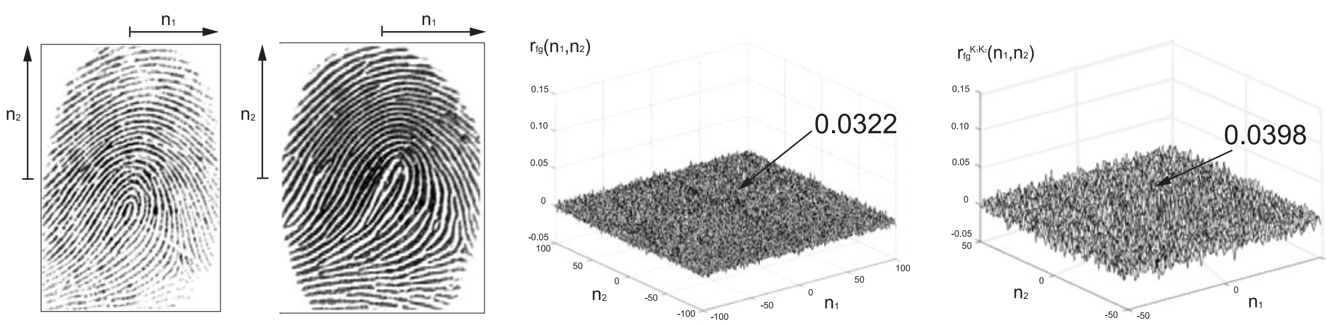
2.2 Fingerprint Matching Algorithm Using BLPOC Function

This section describes a fingerprint matching algorithm using BLPOC function. The algorithm consists of the three



(a) (b) (c) (d)

Fig. 1 Example of genuine matching using the original POC function and the BLPOC function: (a) registered fingerprint image $f(n_1, n_2)$, (b) input fingerprint image $g(n_1, n_2)$, (c) original POC function $r_{fg}(n_1, n_2)$ and (d) BLPOC function $r_{fg}^{K_1K_2}(n_1, n_2)$ with $K_1/M_1 = K_2/M_2 = 0.48$.



(a) (b) (c) (d)

Fig. 2 Example of impostor matching using the original POC function and the BLPOC function: (a) registered fingerprint image $f(n_1, n_2)$, (b) input fingerprint image $g(n_1, n_2)$, (c) original POC function $r_{fg}(n_1, n_2)$ and (d) BLPOC function $r_{fg}^{K_1K_2}(n_1, n_2)$ with $K_1/M_1 = K_2/M_2 = 0.48$.

steps: (i) rotation and displacement alignment, (ii) common region extraction and (iii) matching score calculation with precise rotation.

(i) Rotation and displacement alignment

We need to normalize the rotation and the displacement between the registered fingerprint image $f(n_1, n_2)$ and the input fingerprint image $g(n_1, n_2)$ in order to perform the high-accuracy fingerprint matching. We first normalize the rotation by using a straightforward approach as follows. We first generate a set of rotated images $f_\theta(n_1, n_2)$ of the registered fingerprint $f(n_1, n_2)$ over the angular range $-\theta_p \le \theta \le \theta_p$ with an angle spacing 1°, where bi-cubic interpolation is employed for image rotation. The rotation angle Θ of the input image relative to the registered image can be determined by evaluating the similarity between the rotated replicas of the registered image $f_\theta(n_1, n_2)$ $(-\theta_p \le \theta \le \theta_p)$ and the input image $g(n_1, n_2)$ using the BLPOC function. Next, we align the translational displacement between the rotation-normalized image $f_\Theta(n_1, n_2)$ and the input image $g(n_1, n_2)$. The displacement can be obtained from the peak location of the BLPOC function between $f_\Theta(n_1, n_2)$ and $g(n_1, n_2)$. Thus, we have normalized versions of the registered image and the input image, which are denoted by $f'(n_1, n_2)$ and $g'(n_1, n_2)$.

(ii) Common region extraction

Next step is to extract the overlapped region (intersection) of the two images $f'(n_1, n_2)$ and $g'(n_1, n_2)$. This process improves the accuracy of fingerprint matching, since the non-overlapped areas of the two images become uncorrelated noise components in the BLPOC function. In order to detect the effective fingerprint areas in the registered image $f'(n_1, n_2)$ and the input image $g'(n_1, n_2)$, we examine the n_1-axis projection and the n_2-axis projection of pixel values. Only the common effective image areas, $f''(n_1, n_2)$ and $g''(n_1, n_2)$, with the same size are extracted for the use in succeeding image matching step.

(iii) Matching score calculation with precise rotation

The phase-based image matching is highly sensitive to image rotation. Hence, we calculate the matching score with precise correction of image rotation. We generate a set of rotated replicas $f''_\theta(n_1, n_2)$ of $f''(n_1, n_2)$ over the angular range $-2° \le \theta \le 2°$ with an angle spacing 0.5°, and calculate BLPOC function $r_{f''_\theta g''}^{K_1K_2}(n_1, n_2)$. If the rotation and displacement between two fingerprint images are normalized, the correlation peak can be observed at the center of the BLPOC function. The BLPOC function may give multiple correlation peaks due to elastic fingerprint deformation. Thus, we define the matching score between the two images as the sum of the highest P peaks of the BLPOC function $r^{K_1K_2}_{f''_\theta g''}(n_1, n_2)$, where search area is $B \times B$-pixel block centered at $(0, 0)$. In this paper, we employ the parameters $B = 11$ and $P = 2$. The final score S_{POC} $(0 \leq S_{POC} \leq 1)$ of phase-based matching is defined as the maximum value of the scores computed from BLPOC function $r^{K_1K_2}_{f''_\theta g''}(n_1, n_2)$ over the angular range $-2^\circ \leq \theta \leq 2^\circ$.

3. Feature-Based Fingerprint Matching

We have developed the three different types of feature-based fingerprint matching algorithm: (i) structure matching [4], (ii) string matching [3] and (iii) block matching [8], which are combined with the phase-based image matching to improve the matching performance of fingerprint verification.

In the type of feature-based matching algorithms, minutiae (ridge ending and ridge bifurcation) are used as fingerprint features. In order to extract minutiae, we employ the typical minutia extraction technique [1], which consists of the following four steps: (a) ridge orientation/frequency estimation, (b) fingerprint image enhancement and binarization, (c) ridge thinning, and (d) minutia extraction with spurious minutia removal. Each extracted minutia is characterized by a feature vector $\boldsymbol{m}_i$, whose elements are its (n_1, n_2) coordinates, the orientation of the ridge on which it is detected, and its type (i.e., ridge ending or ridge bifurcation). Let $\boldsymbol{M}^f$ and $\boldsymbol{M}^g$ be sets of minutia feature vectors extracted from the registered image $f(n_1, n_2)$ and the input image $g(n_1, n_2)$, respectively. Figure 3 illustrates an example of minutia extraction.

3.1 Structure Matching

A minutia matching technique based on both the local and global structures of minutiae is used to compute the similarity between $f(n_1, n_2)$ and $g(n_1, n_2)$ [4]. For every minutia $\boldsymbol{m}_i$, we calculate a local structure feature vector $\boldsymbol{l}_i$, which described by the distances, ridge-counts, directions and radial angles of the minutiae relative to each of two nearest-neighbor minutiae and the types of these minutiae. Let $\boldsymbol{L}^f$ and $\boldsymbol{L}^g$ be sets of local structure feature vectors calculated

(a) (b)

Fig. 3 Example of minutia extraction: (a) original fingerprint image, (b) thinned fingerprint image and extracted minutiae (○: ridge ending, □: ridge bifurcation).

from $\boldsymbol{M}^f$ and $\boldsymbol{M}^g$, respectively. We perform minutia matching between $\boldsymbol{M}^f$ and $\boldsymbol{M}^g$ by using their local structure information $\boldsymbol{L}^f$ and $\boldsymbol{L}^g$, and find the best matching minutia pair $(\boldsymbol{m}^f_{i_0}, \boldsymbol{m}^g_{j_0})$, which is called *reference minutia pair*. All other minutiae are aligned based on this reference minutia pair by converting their coordinates to the polar coordinate system with respect to the reference minutia. Thus, we have the aligned minutia information $\boldsymbol{M}'^f$ and $\boldsymbol{M}'^g$. For every aligned minutia $\boldsymbol{m}'^f_i \in \boldsymbol{M}'^f$ (or $\boldsymbol{m}'^g_j \in \boldsymbol{M}'^g$), we calculate a global feature vector $\boldsymbol{g}^f_i$ (or $\boldsymbol{g}^g_j$), which is described by the distance, direction and radial angle of the minutiae relative to the reference minutia $\boldsymbol{m}^f_{i_0}$ (or $\boldsymbol{m}^g_{j_0}$). Based on the distance $|\boldsymbol{g}^f_i - \boldsymbol{g}^g_j|$, we can now determine the correspondence between the minutia pair $\boldsymbol{m}'^f_i$ and $\boldsymbol{m}'^g_j$. As a result, we obtain a set of the corresponding minutia pairs between $\boldsymbol{M}'^f$ and $\boldsymbol{M}'^g$ as well as the matching score $S_{\text{structure}}$ $(0 \leq S_{\text{structure}} \leq 1)$ defined as

$$S_{\text{structure}} = \frac{(\text{\# of corresponding minutia pairs})^2}{|\boldsymbol{M}'^f| \times |\boldsymbol{M}'^g|}. \qquad (6)$$

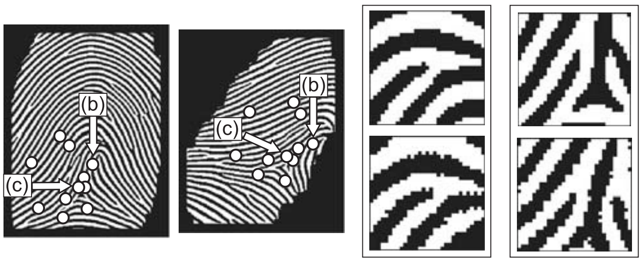
3.2 String Matching

The string matching employs the technique of dynamic programming which can be used to solve the elastic distortion problem in fingerprint matching [3]. We first detect the ridge $\boldsymbol{r}_i$ associated with each minutia $\boldsymbol{m}_i$ during the minutia extraction process. The ridge $\boldsymbol{r}_i$ is represented as a planar curve with its origin corresponding to the minutia and its x-coordinate being the same direction as the minutia direction. The planar curve is also normalized with respect to the average ridge frequency. The rotation angle and translation displacement between $f(n_1, n_2)$ and $g(n_1, n_2)$ are estimated by matching pairs of ridges. The rotation angle and translational displacement which result in the maximum number of matched minutia pairs within a bounding box are considered the correct transformation parameters. Then, the corresponding minutiae are labeled as reference minutia, $\boldsymbol{m}^f_{i_1}$ and $\boldsymbol{m}^g_{i_1}$, respectively. Sets of minutia feature vectors $\boldsymbol{M}^f$ and $\boldsymbol{M}^g$ are converted into polar coordinates with respect to the reference minutia $\boldsymbol{m}^f_{i_1}$ and $\boldsymbol{m}^g_{i_1}$, respectively. The 2D minutia features are reduced to a 1D string by concatenating points in an increasing order of radial angle in polar coordinate. The string matching algorithm is applied to compute the edit distance between the two strings. The corresponding minutia pairs $\boldsymbol{M}''^f$ and $\boldsymbol{M}''^g$ are obtained based on the minimal edit distance between the two strings. The matching score S_{string} $(0 \leq S_{\text{string}} \leq 1)$ is defined as

$$S_{\text{string}} = \frac{(\text{\# of corresponding minutia pairs})^2}{|\boldsymbol{M}''^f| \times |\boldsymbol{M}''^g|}. \qquad (7)$$

3.3 Block Matching

The block matching algorithm extracts the corresponding



(a) (b) (c)

Fig. 4 Example of local block matching using BLPOC function for a genuine pair ($S_{\text{block}} = 0.45$): (a) binarized registered image and binarized input image, (b) a pair of blocks having the highest score (the score of local block matching is 0.45) and (c) a pair of blocks having the lowest score (the score of local block matching is 0.13). The symbols ∘ denote the corresponding minutiae.

minutia pairs between $f(n_1, n_2)$ and $g(n_1, n_2)$, and calculates the matching score by block matching using BLPOC function. We first obtain the corresponding minutia pairs between $f(n_1, n_2)$ and $g(n_1, n_2)$ using the minutia matching, where the structure matching is employed in this paper. When the number of corresponding minutia pairs is greater than 2, we extract local binary images from $f(n_1, n_2)$ and $g(n_1, n_2)$ centered at the corresponding minutia pairs. The size of local binary image is $l \times l$ pixels, where we use $l = 31$ in our experiments. For every pair of local binary images, we align image rotation using the information of minutia orientation, and calculate the BLPOC function between the local image blocks to evaluate the local matching score as its correlation peak value. The score of block matching S_{block} ($0 \leq S_{\text{block}} \leq 1$) is calculated by taking an average of the highest three local matching scores. On the other hand, when the number of corresponding minutia pairs is less than 3, we set $S_{\text{block}} = 0$. Figure 4 shows an example of local block matching using BLPOC function for a genuine pair.

4. Combined Algorithm

This section describes the combined fingerprint matching algorithm proposed in this paper (Fig. 5). In order to combine fingerprint matching algorithms, we employ the score-level fusion techniques such as (i) min rule, (ii) max rule, (iii) sum (or mean) rule and (iv) weighted sum rule [14]. Let $S_{Combine}$, S_i and N_m be a combined score, a matching score and the number of matching scores, respectively. The followings are the brief summary of the score-level fusion techniques.

(i) Min rule: The combined score is the minimum score of matching scores, which is defined by

$$S_{Combine} = \min_i \{S_i\} \quad (i = 1, \cdots, N_m). \tag{8}$$

(ii) Max rule: The combined score is the maximum score of matching scores, which is defined by

$$S_{Combine} = \max_i \{S_i\} \quad (i = 1, \cdots, N_m). \tag{9}$$

(iii) Sum (or mean) rule: The combined score is the sum (or mean) of all the scores, which is defined by

$$S_{Combine} = \sum_{i=1}^{N_m} S_i. \tag{10}$$

(iv) Weighted sum rule: The combined score is the weighted sum of all the scores, which is defined by

$$S_{Combine} = \sum_{i=1}^{N_m} w_i S_i, \tag{11}$$

where w_i is the weight for the matching score S_i and $\sum_{i=1}^{N_m} w_i = 1$.

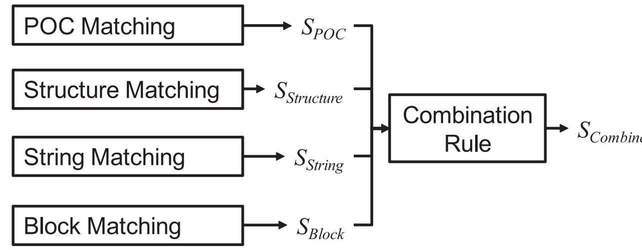


Fig. 5 Score-level fusion of fingerprint matching algorithms.

(a) (b) (c) (d)

Fig. 6 Examples of fingerprint images from DB_A: (a) good-quality fingerprint, (b) dry fingertip, (c) rough fingertip and (d) allergic-skin fingertip.

The combination rules (i)–(iii) are employed to simply combine matching algorithms without learning or optimization process. The weighted sum rule (iv) is employed to evaluate the best performance under linear combination. Although the weighted sum rule (iv) includes the sum rule (iii), we employ the sum rule (iii) to evaluate matching performance of the combined algorithm with simple weights which are not required the optimization process. The weights for the rule (iv) are optimized through a set of experiments.

5. Experiments and Discussion

This section describes a set of experiments using our original database (DB_A) collecting low-quality fingerprint images and the FVC 2002 DB1 set A [20] (DB_B), for evaluating fingerprint matching performance. The following experiments are carried out for the two databases.

- Low-quality fingerprint database (DB_A)
  A set of fingerprint images in this database is captured with a pressure sensitive sensor (BLP-100, BMF Corporation, about 400 dpi) of size 384 × 256 pixels,

(a) (b)

Fig. 7 Examples of genuine pairs from DB_B, which are difficult to verify due to (a) nonlinear distortion and (b) small overlap.

which contains 330 fingerprint images from 30 different subjects with 11 impressions for each finger. In the captured images, 20 of subjects have good-quality fingerprints and the remaining 10 subjects have low-quality fingerprints due to dry fingertips (6 subjects), rough fingertips (2 subjects) and allergic-skin fingertips (2 subjects). Figure 6 shows some examples of fingerprint images. Thus, the test set considered here is specially designed to evaluate the performance of fingerprint matching under difficult condition. We first evaluate genuine matching scores for all the possible combinations of genuine attempts; the number of attempts is ${}_{11}C_2 \times 30$ = 1,650. Next, we evaluate impostor matching scores for impostor attempts: the number of attempts is ${}_{30}C_2$ = 435, where we select a single image (the first image) for each fingerprint and make all the possible combinations of impostor attempts.

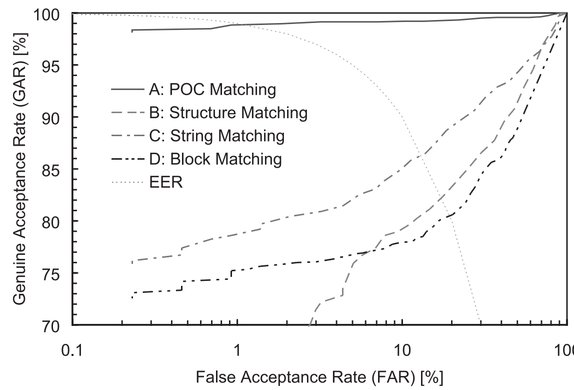
- FVC 2002 DB1 set A (DB_B)
  A set of fingerprint images in this database is captured with an optical sensor (Touch View II, Identx Incorporated, 500 dpi) of size 388×374 pixels, which contains 800 fingerprint images from 100 different subjects with 8 impressions for each finger. Figure 7 shows two examples of genuine pairs, which are difficult to verify due to nonlinear distortion and small overlap. We first evaluate genuine matching scores for all the possible combinations of genuine attempts; the number of attempts is ${}_{8}C_2 \times 100$ = 2,800. Next, we evaluate impostor matching scores for impostor attempts: the number of attempts is ${}_{100}C_2$ = 4,950, where we select a single image (the first image) for each fingerprint and make all the possible combinations of impostor attempts.

The parameters K_1/M_1 and K_2/M_2 of BLPOC function depend on DPI (Dots Per Inch) of fingerprint images. In our experiments, the parameters of BLPOC function are $K_1/M_1 = K_2/M_2 = 0.40$ for DB_A and $K_1/M_1 = K_2/M_2 = 0.48$ for DB_B.

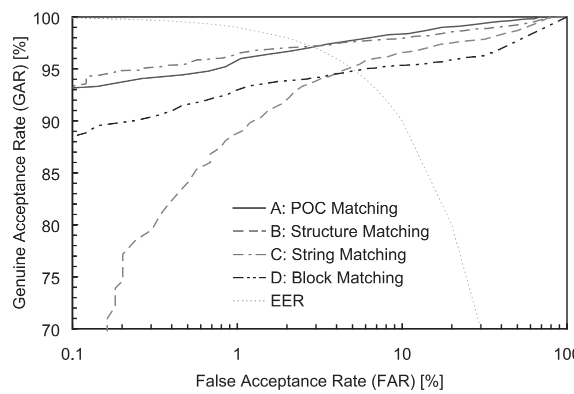
5.1 Performance Evaluation for Individual Algorithms

We compare four fingerprint matching algorithms: (A) a POC-based matching, (B) a structure matching, (C) a string matching and (D) a block matching.

The performance of the biometrics-based verification system is evaluated by the Receiver Operating Characteristic (ROC) curve, which illustrates the Genuine Acceptance Rate (GAR) against the False Acceptance Rate (FAR) at different thresholds on the matching score. The Equal Error



(a)



(b)

Fig. 8 ROC curves and EERs: (a) DB_A and (b) DB_B.

Table 1 EERs for individual algorithms.

Algorithm	DB_A	DB_B
A	1.15%	3.06%
B	18.06%	5.57%
C	13.72%	2.89%
D	19.65%	5.27%

Rate (EER) is also used to summarize performance of a verification system. The EER is defined as the error rate where $100 - \text{GAR} = \text{FAR}$.

Figures 8(a) and (b) show the ROC curves for the four algorithms (A)–(D) for DB_A and DB_B, respectively, while Tables 1(a) and (b) summarize EERs for DB_A and DB_B, respectively. As for DB_A, the POC-based algorithm (A) exhibits significantly higher performance compared with feature-based algorithms (B)–(D), since there are many degraded fingerprint images in DB_A, from which we cannot extract correct minutiae. As for DB_B, the matching performance of the feature-based algorithms (B)–(D) is higher than that for DB_A, since the minutiae are correctly extracted from fingerprint images in DB_B compared with those in DB_A. From these experimental results, the POC-based fingerprint matching algorithm is robust against degraded fingerprint images, while the feature-based fingerprint matching algorithms are robust against distorted fingerprint images.

We employ the correlation coefficient for the quantitative evaluation of the combination of fingerprint matching algorithms. In other words, using the correlation coefficient, we evaluate whether the combination of matching algorithms plays a complementary role in a fingerprint matching task. The correlation coefficient between matching scores can be used to measure the strength of a relationship between a pair of matching algorithms. Table 2 shows the correlation coefficients for all the possible pairs of four matching algorithms. The combinations of the POC-based algorithm (A) and the feature-based algorithms (B)–(D) exhibit low value of the correlation coefficient, that is, each algorithm employs different matching criteria to compute a matching score. On the other hand, the combinations of the feature-based algorithms (B)–(D) exhibit high value of the correlation coefficient, that is, each algorithm employs almost the same matching criteria. As is observed in the above experiment, the combination of the POC-based matching algorithm and the feature-based matching algorithm is expected to improve recognition performance of fingerprint matching compared with individual matching algorithms.

Table 2 Correlation coefficients between all the possible pairs of fingerprint matching algorithms.

Combination	DB_A	DB_B
A, B	0.3179	0.0438
A, C	0.3678	0.0678
A, D	0.3422	0.1391
B, C	0.7475	0.8413
B, D	0.6308	0.7679
C, D	0.5917	0.7505

5.2 Performance Evaluation for Combined Algorithm

We evaluate the matching performance of the combined algorithm described in Sect. 4. In addition to the combination rules mentioned in Sect. 4, we consider the product of some matching scores as shown in Tables 3 and 4, where A×B indicates the product of matching scores between Algorithm (A) and Algorithm (B). The weights for the weighted sum rule are optimized in the sense of EER. In order to determine the optimal values of weights, we evaluate EERs for all the combinations by changing the weight for each matching score from 0.00 to 1.00 at intervals of 0.05, where the sum of weights is always 1. Hence, the total number of weight patterns which we consider in this optimization process is 4,473 patterns for all the matching score combinations.

Tables 3 and 4 show EERs of combined algorithms for DB_A and DB_B, respectively. As for DB_A, some combined algorithms which matching score is calculated using the max and sum rules exhibit improved performance compared with individual matching algorithms. The combinations which improve matching performance are based on the POC-based algorithm and some of feature-based algorithms. The lowest EER for DB_A is 0.65% when using the weighted sum rule, where the weight is 0.75 for (A), 0.05 for (B)×(D) and 0.20 for (C). As for DB_B, some combined algorithms using sum rules exhibit good matching performance. The combinations which improve matching performance are also based on the POC-based algorithm and some of feature-based algorithms. The lowest EER for

Table 3 EERs of combined algorithms for DB_A.

Combination	Min	Max	Sum	Weighted sum	(weight)
A, B	18.06	1.12	1.95	0.94	(0.85, 0.15)
A, C	13.60	0.71	1.01	0.68	(0.70, 0.30)
A, D	12.51	2.77	4.93	1.15	(0.90, 0.10)
B, C	17.46	13.57	13.36	13.01	(0.30, 0.70)
B, D	18.70	19.53	19.03	17.52	(0.90, 0.10)
C, D	17.20	17.05	16.64	13.33	(0.90, 0.10)
A, B×C	14.61	1.01	0.77	0.77	(0.50, 0.50)
A, B×D	18.56	1.01	0.91	0.80	(0.60, 0.40)
A, C×D	17.73	0.97	0.97	0.80	(0.60, 0.40)
A×B, C	6.52	13.45	10.35	5.43	(0.95, 0.05)
A×B, D	12.04	19.65	18.14	9.06	(0.95, 0.05)
A×C, B	3.87	14.31	10.35	2.89	(0.95, 0.05)
A×C, D	11.86	19.38	16.37	3.66	(0.95, 0.05)
A×D, B	14.51	11.80	11.47	11.01	(0.65, 0.35)
A×D, C	12.21	10.86	8.02	7.47	(0.55, 0.45)
B, C×D	17.70	17.55	16.35	16.26	(0.40, 0.60)
B×C, D	17.38	19.65	19.53	16.29	(0.95, 0.05)
B×D, C	18.56	13.69	13.01	12.66	(0.65, 0.35)
A, B×C×D	17.62	1.15	0.94	0.77	(0.35, 0.65)
A×B×C, D	12.26	19.65	19.44	15.93	(0.95, 0.05)
A×B×D, C	12.58	13.72	12.66	9.19	(0.95, 0.05)
A×C×D, B	11.61	18.03	16.17	11.66	(0.95, 0.05)
A, B, C	17.46	0.71	1.45	0.77	(0.60, 0.05, 0.35)
A, B, D	18.70	2.77	5.13	1.13	(0.70, 0.20, 0.10)
A, C, D	17.20	2.51	3.43	0.94	(0.65, 0.25, 0.10)
B, C, D	19.15	16.99	16.58	12.69	(0.25, 0.70, 0.05)
A, B, C×D	17.70	1.12	1.86	0.78	(0.75, 0.10, 0.15)
A, B×C, D	17.38	2.77	4.93	1.01	(0.55, 0.40, 0.05)
A, B×D, C	18.56	0.71	1.21	0.65	(0.75, 0.05, 0.20)
A×B, C, D	12.04	17.05	15.90	9.74	(0.55, 0.40, 0.05)
A×C, B, D	12.21	19.18	16.37	5.90	(0.85, 0.05, 0.10)
A×D, B, C	14.69	11.33	9.71	7.44	(0.60, 0.05, 0.35)
A, B, C, D	19.15	2.51	3.78	0.94	(0.55, 0.10, 0.30, 0.05)

Table 4 EERs of combined algorithms for DB_B.

Combination	Min	Max	Sum	Weighted sum	(weight)
A, B	4.62	1.96	1.49	1.36	(0.55, 0.45)
A, C	2.78	1.32	1.12	1.10	(0.65, 0.35)
A, D	5.03	2.10	2.20	1.26	(0.75, 0.25)
B, C	4.02	2.77	2.53	2.30	(0.35, 0.65)
B, D	4.57	5.18	4.50	4.31	(0.75, 0.25)
C, D	4.56	3.64	3.36	2.35	(0.70, 0.30)
A, B×C	3.17	2.70	1.36	0.78	(0.10, 0.90)
A, B×D	4.41	2.75	1.05	0.86	(0.35, 0.65)
A, C×D	4.14	2.29	1.06	0.85	(0.25, 0.75)
A×B, C	2.45	2.78	2.28	1.44	(0.85, 0.15)
A×B, D	3.66	5.27	4.86	2.63	(0.95, 0.05)
A×C, B	1.78	4.28	2.99	1.14	(0.95, 0.05)
A×C, D	3.78	5.14	4.28	1.35	(0.95, 0.05)
A×D, B	3.86	5.13	4.03	3.51	(0.80, 0.20)
A×D, C	3.89	2.69	1.99	1.88	(0.60, 0.40)
B, C×D	4.20	5.21	4.30	3.65	(0.15, 0.85)
B×C, D	3.85	5.27	4.83	3.05	(0.95, 0.05)
B×D, C	4.49	2.86	2.33	2.06	(0.80, 0.20)
A, B×C×D	3.75	3.04	1.75	0.71	(0.05, 0.95)
A×B×C, D	3.40	5.27	5.09	3.75	(0.95, 0.05)
A×B×D, C	3.63	2.89	2.63	1.78	(0.95, 0.05)
A×C×D, B	3.80	5.56	4.87	3.49	(0.95, 0.05)
A, B, C	4.00	1.33	0.98	0.84	(0.45, 0.30, 0.25)
A, B, D	4.50	2.07	2.17	0.92	(0.55, 0.30, 0.15)
A, C, D	4.61	1.75	1.36	0.84	(0.60, 0.30, 0.10)
B, C, D	4.63	3.56	3.18	2.14	(0.25, 0.65, 0.10)
A, B, C×D	4.20	1.91	0.98	0.75	(0.30, 0.15, 0.55)
A, B×C, D	3.85	2.10	1.96	0.69	(0.20, 0.75, 0.05)
A, B×D, C	4.49	1.32	0.88	<u>0.61</u>	(0.30, 0.55, 0.15)
A×B, C, D	3.66	3.64	3.14	1.81	(0.60, 0.30, 0.10)
A×C, B, D	3.78	5.03	3.81	1.46	(0.80, 0.10, 0.10)
A×D, B, C	3.92	2.75	2.00	1.76	(0.45, 0.15, 0.40)
A, B, C, D	4.60	1.78	1.38	0.67	(0.45, 0.20, 0.25, 0.10)

DB_B is 0.61% when using the weighted sum rule, where the weight is 0.30 for (A), 0.55 for (B)×(D) and 0.15 for (C). As mentioned above, the weighted sum rule needs the time-consuming optimization process for each database. Instead of the weighted sum rule, we can employ the sum rule to obtain relatively good performance. The best EERs using the sum rule are 0.77% for DB_A and 0.88% for DB_B, respectively, which are comparable with the use of the weighted sum rule.

Our observation indicates that the combination of the correlation-based matching and the feature-based matching is effective for improving matching performance, since these algorithms play a complementary role for fingerprint matching tasks.

6. Conclusion

This paper has proposed a novel fingerprint recognition algorithm, which is based on the combination of two different matching criteria: (i) phase-based matching and (ii) feature-based matching. Experimental results clearly show good recognition performance of the combination of the POC-based and feature-based fingerprint matching algorithms compared with each individual algorithm. In our previous work, we have already developed commercial fingerprint verification units for access control applications [10], which employs specially designed ASIC [21], [22] for real-time phase-based image matching. The algorithm in this paper could be easily mapped onto our prototype hardware, since the computational complexity of feature-based matching algorithm is not significant. Prototype hardware implementation and its performance evaluation will be reported in near future.

References

[1] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar, Handbook of Fingerprint Recognition, Springer, 2003.

[2] J. Wayman, A. Jain, D. Maltoni, and D. Maio, Biometric Systems, Springer, 2005.

[3] A.K. Jain, L. Hong, S. Pankanti, and R. Bolle, “An identity-authentification system using fingerprints,” Proc. IEEE, vol.85, no.9, pp.1365–1388, Sept. 1997.

[4] X. Jiang and W.Y. Yau, “Fingerprint minutiae matching based on the local and global structures,” Proc. Int. Conf. Pattern Recognition, vol.2, pp.1038–1041, Sept. 2000.

[5] K. Venkataramani and B.V.K. Vijayakumar, “Fingerprint verification using correlation filters,” Lect. Notes Comput. Sci. (AVBPA2003), vol.2688, pp.886–894, June 2003.

[6] K. Ito, H. Nakajima, K. Kobayashi, T. Aoki, and T. Higuchi, “A fingerprint matching algorithm using phase-only correlation,” IEICE Trans. Fundamentals, vol.E87-A, no.3, pp.682–691, March 2004.

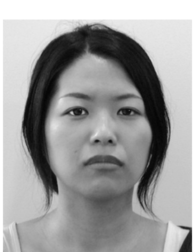
[7] K. Ito, A. Morita, T. Aoki, T. Higuchi, H. Nakajima, and K. Kobayashi, “A fingerprint recognition algorithm using phase-based image matching for low-quality fingerprints,” Proc. IEEE Int. Conf. Image Processing, pp.II–33– II–36, Sept. 2005.

[8] K. Ito, A. Morita, T. Aoki, T. Higuchi, H. Nakajima, and K. Kobayashi, “A fingerprint recognition algorithm combining phase-based image matching and feature-based matching,” Lect. Notes Comput. Sci. (ICB2006), vol.3832, pp.316–325, Dec. 2005.

[9] H. Nakajima, K. Kobayashi, M. Morikawa, A. Katsumata, K. Ito, T. Aoki, and T. Higuchi, “Fast and robust fingerprint identification algorithm and its application to residential access control products,” Lect. Notes Comput. Sci. (ICB2006), vol.3832, pp.326–333, Dec. 2005.

[10] Products using phase-based image matching, http://www.aoki.ecei.tohoku.ac.jp/research/poc.html, Feb. 2001.
[11] C.D. Kuglin and D.C. Hines, "The phase correlation image alignment method," Proc. Int. Conf. Cybernetics and Society, pp.163–165, 1975.
[12] K. Takita, T. Aoki, Y. Sasaki, T. Higuchi, and K. Kobayashi, "High-accuracy subpixel image registration based on phase-only correlation," IEICE Trans. Fundamentals, vol.E86-A, no.8, pp.1925–1934, Aug. 2003.
[13] K. Takita, M.A. Muquit, T. Aoki, and T. Higuchi, "A sub-pixel correspondence search technique for computer vision applications," IEICE Trans. Fundamentals, vol.E87-A, no.8, pp.1913–1923, Aug. 2004.
[14] J. Kittler, M. Hatef, R. Duin, and J. Matas, "On combining classifiers," IEEE Trans. Pattern Anal. Mach. Intell., vol.20, no.3, pp.226–239, 1998.
[15] A.A. Ross, K. Nandakumar, and A. Jain, Handbook of Multibiometrics, Springer, 2006.
[16] S. Parabhakar and A.K. Jain, "Decision-level fusion in fingerprint verification," Pattern Recognit., vol.35, pp.861–874, 2002.
[17] A. Ross, A. Jain, and J. Reisman, "A hybrid fingerprit matcher," Pattern Recognit., vol.36, pp.1661–1673, 2003.
[18] J. Fierrez-Aguilar, L. Nanni, J. Ortega-Garcia, R. Cappelli, and D. Maltoni, "Combining multiple matchers for fingerprint verification: A case study in FVC2004," Lect. Notes Comput. Sci. (ICIAP2005), vol.3617, pp.1035–1042, 2005.
[19] http://bias.csr.unibo.it/fvc2004/
[20] Fingerprint Verification Competition (FVC) 2002, http://bias.csr.unibo.it/fvc2002/, Jan. 2002.
[21] M. Morikawa, A. Katsumata, and K. Kobayashi, "An image processor implementing algorithms using characteristics of phase spectrum of two-dimensional fourier transformation," Proc. IEEE Int. Symp. Industrial Electronics, vol.3, pp.1208–1213, 1999.
[22] M. Morikawa, A. Katsumata, and K. Kobayashi, "Pixel-and-column pipeline architecture for FFT-based image processor," Proc. IEEE Int. Symp. Circuit and Systems, vol.3, pp.687–690, 2002.

Koichi Ito received the B.E. degree in electronic engineering, and the M.S. and Ph.D. degree in information sciences from Tohoku University, Sendai, Japan, in 2000, 2002 and 2005, respectively. He is currently an Assistant Professor of the Graduate School of Information Sciences at Tohoku University. From 2004 to 2005, he was a Research Fellow of the Japan Society for the Promotion of Science. His research interest includes signal and image processing, and biometric authentication.



Ayumi Morita received the B.E. degree in information, and the M.S. degree in information sciences from Tohoku University, Sendai, Japan, in 2004 and 2006, respectively. She is currently with Systems Development Department, Yamatake Corporation, Fujisawa, Japan. Her research interest includes biometric image processing.

Takafumi Aoki received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1988, 1990, and 1992, respectively. He is currently a Professor of the Graduate School of Information Sciences at Tohoku University. For 1997–1999, he also joined the PRESTO project, Japan Science and Technology Corporation (JST). His research interests include theoretical aspects of computation, VLSI computing structures for signal and image processing, multiple-valued logic, and biomolecular computing. Dr. Aoki received the Outstanding Paper Award at the 1990, 2000, 2001 and 2006 IEEE International Symposiums on Multiple-Valued Logic, the Outstanding Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1989 and 1997, the IEE Ambrose Fleming Premium Award in 1994, the IEICE Inose Award in 1997, the IEE Mountbatten Premium Award in 1999, the Best Paper Award at the 1999 IEEE International Symposium on Intelligent Signal Processing and Communication Systems, the IP Award at the 7th LSI IP Design Award in 2005, and the Best Paper Award at the 14th Workshop on Synthesis And System Integration of Mixed Information technologies.

Hiroshi Nakajima received the B.E. degree in electronic engineering from Tohoku University, Sendai, Japan, in 1990. He is currently with Systems Development Department, Yamatake Corporation, Fujisawa, Japan. His research interest includes biometric image processing.

Koji Kobayashi received the B.E. and M.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1976, and 1978, respectively. He is currently a general manager of Vision Sensing Department, Yamatake Corporation, Fujisama, Japan. His general interests includes real-time automation system architecture, network communication protocol LSI, biometric image processing, CMOS image sensor, and three-dimensional sensing.

Tatsuo Higuchi received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1962, 1964, and 1969, respectively. He is currently a Professor at Tohoku Institute of Technology and a Emeritus Professor at Tohoku University. From 1980 to 1993, he was a Professor in the Department of Electronic Engineering at Tohoku University. He was a Professor from 1994 to 2003, and was Dean from 1994 to 1998 in the Graduate School of Information Sciences at Tohoku University. His general research interests include the design of 1-D and multi-D digital filters, linear time-varying system theory, fractals and chaos in digital signal processing, VLSI computing structures for signal and image processing, multiple-valued ICs, multiwave opto-electronic ICs, and biomolecular computing. Dr. Higuchi received the Outstanding Paper Awards at the 1985, 1986, 1988, 1990, 2000, 2001 and 2006 IEEE International Symposiums on Multiple-Valued Logic, the Certificate of Appreciation in 2003 and the Long Service Award in 2004 on Multiple Valued Logic, the Outstanding Transactions Paper Award from the Society of Instrument and Control Engineers (SICE) of Japan in 1984, the Technically Excellent Award from SICE in 1986, and the Outstanding Book Award from SICE in 1996, the Outstanding Transactions Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 1990 and 1997, the Inose Award from IEICE in 1997, the Technically Excellent Award from the Robotics Society of Japan in 1990, the IEE Ambrose Fleming Premium Award in 1994, the Outstanding Book Award from the Japanese Society for Engineering Education in 1997, the Award for Persons of scientific and technological merits (Commendation by the minister of state for Science and Technology), the IEE Mountbatten Premium Award in 1999, the Best Paper Award at the 1999 IEEE International Symposium on Intelligent Signal Processing and Communication Systems and and the Best Paper Award at the Workshop on Synthesis And System Integration of Mixed Information technologies (SASIMI) in 2007. He also received the IEEE Third Millennium Medal in 2000. He received the fellow grade from IEEE and SICE.