INVITED PAPER *Special Issue on the 2001 IEICE Excellent Paper Award*

"Smartface"–A Robust Face Recognition System under Varying Facial Pose and Expression*

Osamu YAMAGUCHI† *and* **Kazuhiro FUKUI**†, ***Regular Members***

SUMMARY Face recognition provides an important means for realizing a man-machine interface and security. This paper presents "Smartface," a PC-based face recognition system using a temporal image sequence. The face recognition engine of the system employs a robust facial parts detection method and a pattern recognition algorithm which is stable against variations of facial pose and expression. The functions of Smartface include (i) screensaver with face recognition, (ii) customization of PC environment, and (iii) real-time disguising, an entertainment application. The system is operable on a portable PC with a camera and is implemented only with software; no image processing hardware is required.

key words: *face recognition, processing of image sequence, facial pose, variation of facial expression, application using face recognition*

1. Introduction

Recently, personal identification using face recognition has attracted significant interest in the fields of human interface and security. Face recognition has two principal merits: it is a noncontact method and the subject is unaware that recognition is being performed. In addition, it can be utilized for not only personal identification but also for several applications which use the position of the face and facial parts in an image [1].

This paper presents a face recognition system constructed on a portable personal computer (PC). To construct a face recognition system on a portable PC, which has the advantage of easy portability but lacks the powerful calculation capability of a server, it is necessary to consider the following points,

(a) Easing the user's burden with regard to face recognition
(b) Reduction of calculation cost

For point (a), in the construction of an easy-to-use face recognition system, it is important to ensure that the constraints on the user are not severe. A conventional face recognition system (eg. [2]) imposes certain restrictions such as "keep the face straight and maintain a neutral facial expression" during image capture. In order to achieve recognition in the case of natural posture, and thus to avoid invoking a defensive attitude on the part of the subject, it is necessary to accommodate the changing appearance of the subject due to variation of facial expressions and to ease restrictions on face direction. Moreover, face direction tends to change easily in the case of using a portable system because the physical relationship between the user and system involves a large degree of freedom; for instance, the user may hold the system in both hands or place it on his lap.

Given these circumstances, the conventional method based on still image recognition is subject to a high incidence of incorrect recognition [3]. So, in the present work, instead of using still image recognition, the distribution of an input image sequence in which face direction and facial expression may be varied is evaluated. In this system, we implemented the face recognition method using multiple images from an image sequence [3], [4].

This system requires a function which can detect a facial pattern from an arbitrary scene. For this purpose, a detection method for facial parts [5], involving the combination of geometric feature information and pattern information, is applied to the system.

On the other hand, for point (b), considering that the face recognition system is likely to be operating in parallel with other application software, it is necessary that the implementation of the system be such that the calculation cost is as low as possible. Since image processing applications generally involve a high calculation cost and require large memory, for example in comparison to audio processing, it is necessary to reduce the processing cost.

Moreover, in this system, the calculation cost is increased by the use of image sequence processing with an algorithm to lighten the user's burden. Since the target PC has limited calculation power, it is important to reduce the calculation cost. In this study, we investigate the revision of an algorithm for fast computation, the means for tracking the feature points in an image sequence, the active use of multimedia instructions built into processors, and the control of image capture.

Next, we discuss how the result of face recognition is utilized for applications. For instance, with regard to security, measures are implemented to prevent unauthorized users from viewing data saved on the disk on the screen. Also, user identification enables

Manuscript received July 15, 2002.

†The authors are with Toshiba Corporate Research and Development Center, Kawasaki-shi, 212–8582 Japan.

*This article was originally published in IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J84-D-II, no.6, pp.1045–1052, June 2001, and has been revised from the English translation, To appear Systems and Computers in Japan (John Wiley & Sons).

the user to be provided with the services and information corresponding to his needs, and, in terms of the man-machine interface, an environment is set up for the individual user.

The information obtained from face recognition can be used not only for personal identification but also for various other purposes. This paper presents "Smartface," a face recognition software designed for convenient use, which includes an amusement factor. The system's function include: (1) sound response and configuration of PC by personal identification, (2) screensaver with face recognition, and (3) real-time disguising simulation.

In Sect. 2, we present a facial feature point detection method and a face identification algorithm robust against varying facial pose and expression. In Sect. 3 we explain the system architecture and method of reduction of calculation cost. In Sect. 4 we describe the application implemented on the target PC.

2. A Robust Face Recognition Algorithm under Varying Facial Pose and Expression

Figure 1 shows the processing flow. The processing flow which comprises (1) capturing the image from a camera, (2) detection of face region from an input image, (3) detection of facial feature points (pupils and nostrils), (4) pattern cropping in which orientation and size are normalized, (5) calculation of input subspace, and (6) recognition by similarity calculation with a reference dictionary, is performed in order and personal identification is carried out. As a result, (7) applications using face recognition results are processed.

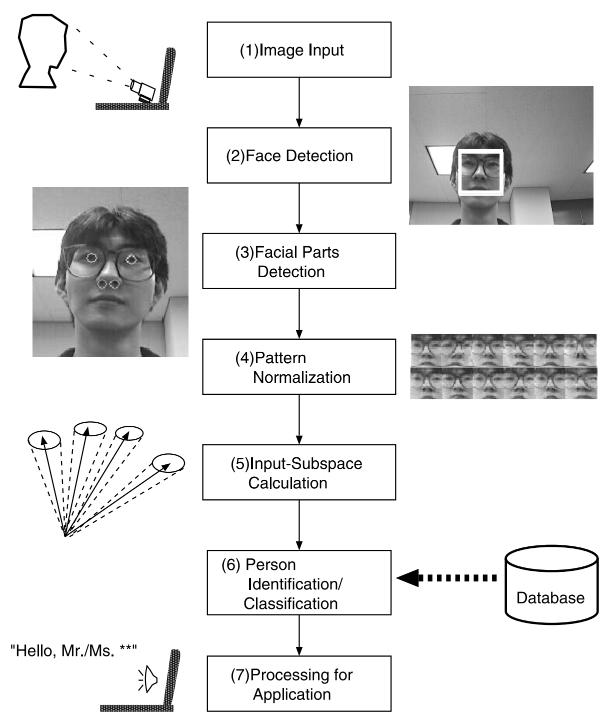


Fig. 1 System flowchart.

2.1 Face Detection

Face detection is based on pattern matching using the subspace method [6]. We collect data on the subject's face (15×15 pixel image) and generate a template dictionary for face detection. Scanning this template for the image, the location of the face is determined as the position which gives the global maximum of similarity to the dictionary while exceeding a threshold. Changing the size of a face region, template matching is executed for multiple scaled input images.

Furthermore, when the system detects the face region from a complex scene, a problem arises in that a non-face region may be mistaken for a face region. To eliminate this problem, using an approach similar to that proposed in [7], we have collected non-face patterns which have a high similarity to face patterns and made a non-face dictionary. Using this dictionary, we calculate the similarity to the non-face dictionary. The detection method determines the correct face position, utilizing two similarities of the face and the non-face dictionary, respectively.

2.2 Detection of Facial Feature Point and Cropping the Pattern

For the detected face region, we apply our feature point detection method [5], to detect the positions of the eye and nose (pupils and nostrils). The detection method executes stable facial feature point detection based on a combination of shape extraction and pattern matching.

First, candidate feature points are extracted as circular regions by a separability filter. The separability filter, illustrated in Fig. 2 (a), has two circular regions. Separability $S (0.0 < S \leq 1.0)$ is calculated and defined by Eq. (1)

$$S = \frac{n_1(\bar{P}_1 - \bar{P}_m)^2 + n_2(\bar{P}_2 - \bar{P}_m)^2}{\sum_{k=1}^{N}(P_k - \bar{P}_m)^2}, \tag{1}$$

where N is the number of pixels in all regions, n_1 and n_2 are the numbers of pixels in regions 1 and 2, P_k is the image intensity at pixel k, $\bar{P_m}$ is the mean of the

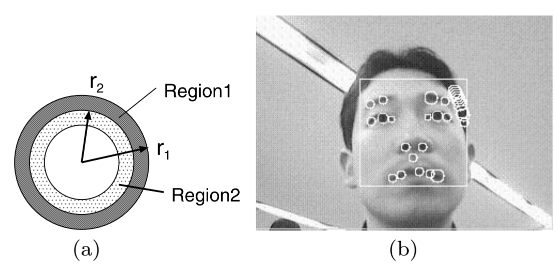


Fig. 2 (a)Separability filter, (b)Candidate feature points.

image intensity for the combined local region, and $\bar{P}_1$ and $\bar{P}_2$ are the means of the image intensity in regions 1 and 2, respectively.

For each pixel in the detected face region, the separability is calculated while changing the position and the radius of a filter. The candidate feature points are selected from some of the local maximum peaks of separability. The candidate feature points are shown in Fig. 2 (b).

The candidates are then verified by the similarity of pattern matching for eye and nose patterns using the subspace method. Finally, four candidates are selected through the constraints that the similarity exceeds the threshold and the four feature points satisfy the geometrical positions of facial parts. The details are described in [5].

Next, a square gray scale pattern is normalized in terms of position and size, and is cropped based on extracted feature points of the eye and nose (pupils and nostrils). Figure 3 shows 20 examples of square gray scale patterns (30 × 30 pixel image). Numerous patterns are generated automatically without human intervention.

2.3 Personal Identification by the Mutual Subspace Method

To alleviate the influence of facial pose and expression, the system applies the recognition method using multiple images from a temporal image sequence [3], [4].

Figure 4 shows the basic concept of the approaches used for still image recognition and image sequence recognition. The diagrams display the distribution of sample reference data of each person with the meshed region. Figure 4 (a) shows the conventional method using a still image. By judging the distance between single input (× mark in the figure) and each sample set, the recognition is performed. In this example, it is difficult to determine which person should be identified.

On the other hand, Fig. 4 (b) shows the multiple image recognition method which uses various multiple input data sequentially. If a distribution (set of × marks in the figure) of the input face data in an image sequence discriminates whether it is close to a distribution of either person, stable recognition despite facial variation is attained.

We introduce the Mutual Subspace Method (MSM) [8], a pattern recognition method using multiple image data. In this method, input vectors are represented as a subspace and it is applied in such a way that the similarity is defined by the angle between the input and the reference subspaces.

Figure 5 illustrates the MSM. The squared cosine of the angle between two subspaces, D and G, is defined as

$$\cos^2\theta = \sup_{d\in D, g\in G, ||d||\neq 0, ||g||\neq 0} \frac{|(d,g)|^2}{||d||^2||g||^2}, \tag{2}$$

where d and g are vectors on the subspaces while Eq. (2) has local maxima. The largest eigenvalue λ_{max} of the matrix PQP yields $\cos^2\theta$, where P and Q are, respectively, the orthogonal projection matrices on the subspaces D and G [8],

$$\cos^2\theta = \lambda_{max}. \tag{3}$$

D denotes the subspace of a reference pattern, and G denotes the subspace of a multiple input pattern. The largest eigenvalue of matrix PQP is not calculated, but, in practice, the eigenvalue problem of matrix X which is denoted by Eq. (4) is solved.

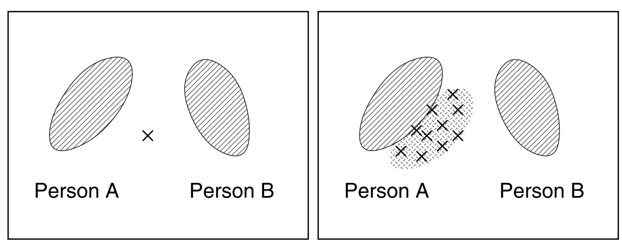
Then the largest eigenvalue is defined as the similarity S_{mutual}, that is *the inter-subspace similarity* [8].

Let matrix X be

$$X = (x_{ij}) = \sum_{m=1}^{M} (\psi_i, \phi_m)(\phi_m, \psi_j), \tag{4}$$

where ϕ_i and ψ_j are the eigenvectors of subspaces D and G, and M and L are the number of dimensions

Fig. 3 Example of normalized patterns.



(a) Still image (single input)

(b) Image Sequence (multiple inputs)

Fig. 4 Recognition using still image and image sequence.

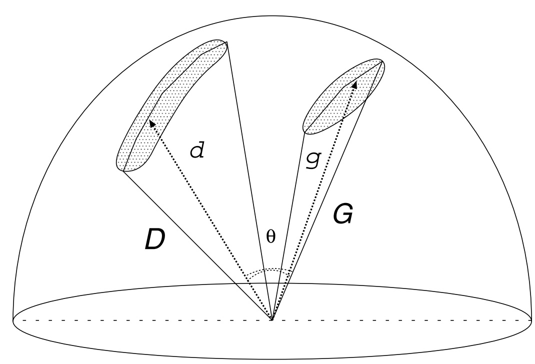


Fig. 5 Mutual subspace method.

Table 1 Performance of identification by the MSM.

Dimension of Reference	1	3	5	7	10
Dimension of Input	1	3	5	7	10
Correct rate	91.1%	96.9%	98.5 %	98.7%	99.0%

of the subspaces D and G (assuming $L \leq M$), respectively.

$$W^T XW = \Lambda, \tag{5}$$

where W is a diagonalization matrix, and λ_{max} is the largest value of diagonal components of matrix Λ. Consequently,

$$S_{mutual}(G, D) = \lambda_{max}. \tag{6}$$

To demonstrate the effectiveness of the proposed method, experiments were performed off-line. We collected faces of 101 individual data, 36360 images containing 180 registration data and 180 test data for each person by two acquisitions. Data were captured while the person turned his or her head in several directions, and changed his or her facial expressions; the data include motions of the mouth in conversation.

First, we performed an identification experiment in which one of the registered categories with the highest similarity to the subject person was identified. On changing the dimension of the input and the reference subspaces as shown in Table 1, the correct identification rate was 99.0% when each dimension was 10. Applying the higher dimensions, the correct identification rate was not improved.

Next, we attempted to carry out a verification experiment in which it is judged whether the subject person is the same person as the candidate person, and the performance was compared with that of conventional methods [3], [4].

Calculating the False Acceptance Rate (FAR) and the False Rejection Rate (FRR) and evaluating by the Equal Error Rate (EER) so that the two error rates are equivalent, EER of the proposed method becomes 3%, whereas that of the conventional method is 8%, when the dimension is 5.

3. System Configuration

3.1 Target System

This system (shown in Fig. 6) was implemented as an application software which was operable on Windows 98 on the Toshiba mini-notebook PC Libretto ff 1100 (CPU: Intel MMX Pentium 266 MHz). The PC is equipped with a CMOS camera (inside the circle in the figure), which is pointing upwards.

Figure 7 shows the software architecture. The engines of face and facial parts detection and personal identification, which are described in Sect. 2, are constructed as a DLL (Dynamic Linkage Library) independently. These face recognition engines are built into the DirectShow filter, which is utilized by Smartface application in the upper layer. Face and facial parts detection DLL and identification DLL use a template dictionary for detection and a personal dictionary for identification, respectively.

Fig. 6 Target PC System.

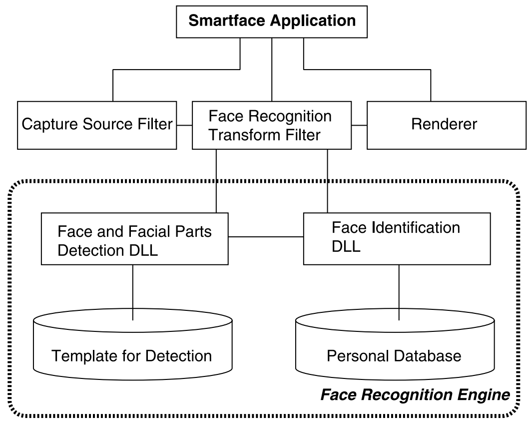


Fig. 7 Software architecture.

3.2 Processing of Identification and Registration

In the identification process, the process shown in Fig. 1 is performed with each image input. A cropped facial pattern is applied for histogram equalization, and subsampled to quarter size. We obtain a feature vector of 225 dimensions (15×15) with image intensity values assigned to the elements. An input subspace is formed from these vector data. We compute the similarity between the input subspace and the registered reference subspaces on a database, and the candidate with the highest similarity is determined as the identified person if the similarity exceeds a threshold.

In the registration stage, we entered the individual name, and captured 50 images. From capturing the image to the calculation of dictionary, the time required for registration is about 20 seconds. When images of the user were captured with variation which is caused by changing facial pose and facial expression, an accurate recognition can be performed later. A maximum of 100 people can be registered in the system.

To allow for a change in the properties of a person with the passage of time, we confirmed that the correct recognition rate was maintained by adding new data to each person's dictionary and updating the dictionary [10]. Since the user can select an updating function of the registration utility in the Smartface application, these data can be added.

3.3 Implementation in Consideration of Computation Cost

This section describes the method of reducing the computation cost of each process in order to realize a practical processing speed.

[Image Capturing]

The size of an input image is 320 × 240 pixels. When the application begins, image data transfer to the memory is always executed. Since the computation cost of face detection is large, if the system could not detect the face region, it decreases the frame rate, the number of frames per second of an input. This means a decrease of the number of frames processed per second and the reduction of CPU usage of the application. Then, face detection is executed intermittently. If the system detects the face again, the recognition is executed at the normal frame rate.

[Face detection and facial parts detection]

Face detection and facial parts detection are not executed simultaneously in every single frame. Face detection is not executed in the frame succeeding a frame in which facial parts detection was correctly executed. Moreover, the search area is limited to the neighborhood of the detected feature points, and the tracking process is executed. The face region detection is activated in the next frame only when the facial feature points fail.

[Computation of input subspace in the personal identification process]

In the registration stage of this system, in the calculation to form the reference subspace, the QR algorithm is executed for the correlation matrix which is accumulated in the multiple vector data. On the other hand, since the calculation to form the input subspace is executed in every frame, use of the same calculation algorithm has a large cost. To realize fast recognition using the MSM, we adopted the Simultaneous Iteration Method in which the subspace is calculated successively [6]. The Simultaneous Iteration Method is advantageous in that the system does not retain all input vectors which are used for computation and calculates only the number of required eigenvectors.

The estimation of the correlation matrix is updated as

$$\hat{R}_{k+1} = (1-\mu)\hat{R}_k + \mu g_{k+1} g_{k+1}^T,$$

where $\hat{R}_k$ is the estimation of the correlation matrix, g_{k+1} is the input vector, and μ is a parameter which has a small value. Let r be the repetition times (the number of input data), and the parameter may be represented as $\mu = \frac{1}{r+1}$.

Next, the matrix C_k which has eigenvectors in a column is updated as

$$\hat{C}_k = \hat{R}_k C_{k-1},$$

$$C_k = GS(\hat{C}_k),$$

where $GS(\cdot)$ is function of Gram-Schmidt orthogonalization.

[Employment of multimedia instruction]

In addition, the multimedia instructions of the CPU are used in each process, and improve the processing speed. Since the vector inner product operation is in heavy usage in this system, the efficiency of this operation leads to reduction of calculation cost.

We examine the effectiveness of introducing multimedia instructions after the integerization of the floating operation. Table 2 shows the measurement result of the computation for the input subspace as mentioned above and the calculation of the MSM by a profiler. These computation costs are reduced to about 1/2 and 1/4, respectively. The employment of integerization is also effective for face detection and facial parts detection which uses the pattern matching based on the subspace method. The computation precision by the integerization affected the error in the third decimal place. However, since the order of similarity in the subspace method does not change, it does not affect the recognition performance.

[Setting up the dimension for similarity calculation in consideration of recognition performance]

While maintaining the recognition performance, we select the dimensions of input and reference subspaces from the viewpoint of reduction of the computation cost of similarity calculation. According to the results in Table 1, comparing the case of the number of dimensions between 7 and 10, the decrease of the recognition performance is small. In addition, the number of inner production operations in Eq. (4) is decreased to 1/3 in the case of 10 dimensions. For these two reasons,

Table 2 Effectiveness using multimedia extension command.

	Without multimedia instruction (float)	With multimedia instruction (short int)
Computation of input subspace	1700	851
Calculation of the MSM	2855	645

(unit : μs)

the number of dimensions is determined as 7.

Using the portable PC practically, the identification test was executed for 50 persons, and the recognition performance was the same as that shown in Table 1. The system can process 5 or 6 frames per second from feature point detection to personal identification.

4. Application Features of Smartface

4.1 Capture Viewer

When the system starts up, the Capture Viewer usually opens as shown in Fig. 8 and displays the input image overlapping the detection result of the face and eye regions. The icon which represents the status of recognition and the graph of the person's similarity give the users feedback on the system status.

4.2 User Customization

User customization is a function of the configuration for each registered user. An audio reply and a starter application can be set. Normally, the name entered when the face was registered is articulated by the speech synthesizer, for example, "Hello, Mr./Ms. ○○○." Any sound source file can be played when the identification is executed. Moreover, if the user has selected an application software in advance, the system can start the application automatically when the identification is executed. For instance, if the user has chosen the application which notifies him if an email has arrived, the system can inform the identified user of the existence of the email.

4.3 Screensaver with Face Recognition

A registered password must be used in order to unlock a conventional screensaver. Smartface screensaver can be unlocked at various times by face recognition. The selectable times are (1) the time when the face is detected, (2) the time when the registered person is identified, and (3) the time when the specific person is identified. These times can be set according to the purpose or the degree of security desired.

Fig. 8 The Capture Viewer window.

4.4 Real-Time Disguising Simulation

Smartface performs not only personal identification but also automatic and realtime disguising simulation using the position and the size of a face region and the positions of the eye and the nose.

This function executes superimposition of bitmaps generated by computer graphics, or applies image processing and deformation of a partial image according to the detected face position and size. Figure 9 shows disguising with "Pierrot." The user can select the various contents by pressing the button of the Capture Viewer.

In the conventional means of ornamenting a face image, the user inputs the positions of the face region and facial parts manually, or adjusts the face position in the image. In the example of automatic detection described in [9], the lighting condition and the background were fixed and only a still image was captured. Then the image frame was ornamented in accordance with the detected face region. Smartface does not have the limitation of a fixed background, and can ornament the image sequence continuously. Figure 10 shows the image sequence captured in front of a complex background by moving the camera from right to left. Though the face region moved, the system tracked it and it could be ornamented.

The contents of disguising are described by script language which designates the deformation and combines the multiple bitmaps. Figure 11 (a) contains the disguising script for "Pierrot," as shown in Fig. 9.

Figure 11 (b) shows the designation of the relative position and size for the bitmap region and deformable region in accordance with the detected facial region. Referring to the script shown in Fig. 11 (a), we explain the vertex P_1 of a rectangle in which the bitmap is arranged. The vertex P_1 is represented by *a reference point vector* denoting the position of detected left eye (LEFT_EYE), two *relative position vectors* denoting the relative position and size based on width and height of detected facial region (FACE_WIDTH, FACE_HEIGHT), and the constant of magnification ratio.

These scripts, as shown in Fig. 11 (a), describe the multiple vertices that determine the arranged position. This description maintains the relation of the relative position and size even if the face position and size change. This script language is used not only for preparing the command for overlapping of the bitmap but also for preparing the command for mosaic pattern and partial pattern scaling. Addition, alternation and editing of contents can be executed freely.

Figure 12 shows examples of images which were applied to various human subjects. This function can also be utilized for preprocessing of image telecommunication using MPEG4, and so forth.

5. Conclusion

This paper proposed the face recognition system “Smartface” which can operate on a portable PC.

Since in the case of using a portable PC, the ability to cope with variation of face direction and facial expression is indispensable, we introduced a recognition method using an image sequence. However, the introduction of the recognition using an image sequence led to an increase in the computation cost. We solved the problem by using feature point tracking and by improving the calculation algorithm. As a result, we realized a face recognition system that achieves high recognition performance while, at the same time, lightening the user’s burden.

In future studies, we intend to develop a further robust face and facial parts detection method, and to implement a recognition method which is robust against

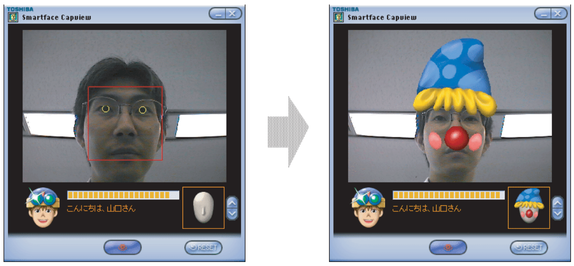
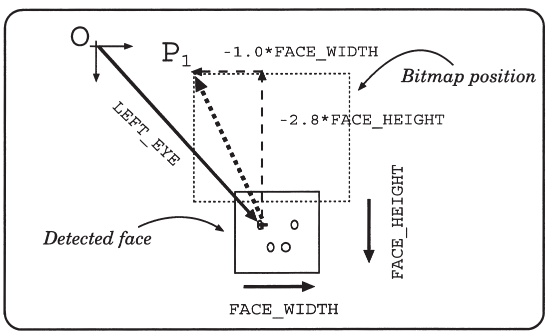


Fig. 9 Simulation of disguising.

```
# Script for "pierrot"
FG_EMBEDBMP 2
LEFT_EYE ADD FACE_WIDTH -1.0 ADD FACE_HEIGHT -2.8
RIGHT_EYE ADD FACE_WIDTH 1.0 ADD FACE_HEIGHT 0
COLORBMP "data/pierrot1.bmp" 255 255 255

FG_EMBEDBMP 2
CENTER_NOSE ADD FACE_WIDTH -0.5 ADD FACE_HEIGHT -0.5
CENTER_NOSE ADD FACE_WIDTH 0.5 ADD FACE_HEIGHT 0.5
COLORBMP "data/pierrot2.bmp" 255 255 255

FG_EMBEDBMP 2
RIGHT_EYE ADD FACE_WIDTH -0.1 ADD FACE_HEIGHT 0.4
RIGHT_EYE ADD FACE_WIDTH 0.5 ADD FACE_HEIGHT 1.2
COLORBMP "data/pierrot3.bmp" 255 255 255

FG_EMBEDBMP 2
LEFT_EYE ADD FACE_WIDTH -0.5 ADD FACE_HEIGHT 0.4
LEFT_EYE ADD FACE_WIDTH 0.1 ADD FACE_HEIGHT 1.2
COLORBMP "data/pierrot4.bmp" 255 255 255
```

P_1

(a)



(b)

Fig. 11 (a)Script for disguising “Pierrot,” (b)Designation of relative position.

Fig. 12 Example of disguising simulation.

Fig. 10 Processing of Image Sequence.

changes in the lighting condition as well. Furthermore, we would like to develop a face recognition application for small mobile appliances other than PCs.

Acknowledgments

The authors would like to thank Ken-ichi Maeda, chief research scientist of Toshiba Corporate Research and Development Center, for his valuable comments on the application of his MSM to face recognition. They are also indebted to the members of the Multimedia Laboratory, Toshiba Corporate Research and Development Center, for their support and comments regarding this work.

References

[1] O. Hasegawa, S. Morishima, and M. Kaneko, "Processing of facial information by computer," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J80-D-II, no.8, pp.2047–2065, Aug. 1997.
[2] M. Doi, K. Sato, and K. Chihara, "A robust face identification against lighting fluctuation for lock control," Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp.42–47, April 1998.
[3] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition system using temporal image sequence," IEICE Technical Report, PRMU97-50, 1997.
[4] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition using temporal image sequence," Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition, pp.318–323, April 1998.
[5] K. Fukui and O. Yamaguchi, "Facial feature point extraction method based on combination of shape extraction and pattern matching," Systems and Computers in Japan, vol.29, no.6, pp.49–58, 1998.
[6] E. Oja, Subspace Methods of Pattern Recognition, Research Studies Press, England, 1983.
[7] K.K. Sung and T. Poggio, "Example-based learning for view-based human face detection," IEEE Trans. Pattern Anal Mach. Intell., vol.20, no.1, pp.39–51, 1998.
[8] K. Maeda and S. Watanabe, "A pattern matching method with local structure," IEICE Trans., Inf. & Syst. (Japanese Edition), vol.J68-D, no.3, pp.345–352, March 1985.
[9] N. Tabata, "Face research on amusement," IEICE Human Communication Technical Seminar, Proc. Face Research for Human Communication, March 1999.
[10] O. Yamaguchi and K. Fukui, "Updating face dictionary against temporal change," IEICE Technical Report, PRMU99-25, 1999.
[11] K. Fukui, O. Yamaguchi, K. Suzuki, and K. Maeda, "Face recognition under variable lighting condition with constrained mutual subspace method – Learning of constraint subspace to reduce influence of lighting changes," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J82-D-II , no.4, pp.613–620, April 1999.

Osamu Yamaguchi received his B.E. and M.E. degrees from Okayama University, in 1992 and 1994, respectively. In 1994, he joined Toshiba Corporation. He is currently a research scientist at Multimedia Laboratory, Toshiba Corporate Research and Development Center. He is a member of IPSJ.

Kazuhiro Fukui received his B.E. and M.E. degrees from Kyushu University, in 1986 and 1988, respectively. In 1988, he joined Toshiba Corporation. He is currently a senior research scientist at Multimedia Laboratory, Toshiba Corporate Research and Development Center.