Information Sciences 269 (2014) 1–20

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Two-factor face authentication using matrix permutation transformation and a user password ☆

Jeonil Kang [a], DaeHun Nyang [b,*], KyungHee Lee [c]

[a] School of Information and Communication Engineering, INHA University, Incheon, Republic of Korea
[b] School of Computer and Information Engineering, INHA University, Incheon, Republic of Korea
[c] Department of Electrical Engineering, University of Suwon, Suwon, Republic of Korea

ARTICLE INFO

Article history:
Received 24 December 2010
Received in revised form 27 June 2013
Accepted 1 February 2014
Available online 12 February 2014

Keywords:
Face authentication
Biometrics security
User privacy

ABSTRACT

Although authentication using biometric techniques is convenient, security issues such as the loss of personal bio-information are serious problems. However, the development of a secure biometrics scheme poses considerable challenges because users' bio-information is not precisely the same for each authentication attempt. This uncertainty during the authentication process obstructs direct application of cryptographic one-way functions in the authentication system. In this paper, we suggest a two-factor face authentication scheme using matrix transformations and a user password. Our scheme is designed with a secure cancellation feature, in that templates composed of permutation and feature vectors can be freely changed. Through experimental scenarios and results, we introduce the notable features of our scheme. Furthermore, we consider possible attacks on the proposed scheme and suggest security enhancement methods.

1. Introduction

A one-way transformation strengthens the security of encoded information if, upon comparison, the results of the transformed domain correspond with those in the original domain. On this basis, several authentication schemes have been introduced to adopt this unidirectional feature in their security strategies. Generally, unidirectionality makes template cancellation possible in biometrics. An old template should be replaced by a new template only after cancellation of the old template if security issues arise. Without unidirectionality, an attacker could recover bio-information from corresponding templates even after cancellation.

Owing to the difficulties encountered with unidirectionality in biometrics, relatively few schemes have been proposed. In particular, in face authentication, developing a scheme that has a high security level with high accuracy is difficult because the features used for classifying each face are not easily defined. In brief, sufficiently good classifiers are lacking in the field of face authentication, as opposed to other biometrics areas. Nevertheless, extensive research on face authentication, including security challenges, has been performed, because face authentication has considerable advantages in terms of both economics and non-intrusiveness.

☆ Portions of the research in this paper use the FERET database of facial images collected under the FERET program, sponsored by the DOD Counterdrug Technology Development Program Office.

* Corresponding author. Address: #307 Hi-tech Center, INHA University, 100 Inha-ro, Nam-gu, Incheon 402-751, Republic of Korea. Tel.: +82 32 876 8424; fax: +82 32 865 0480.

E-mail addresses: dreamx@isrl.kr (J. Kang), nyang@inha.ac.kr (D. Nyang), khlee@suwon.ac.kr (K. Lee).

http://dx.doi.org/10.1016/j.ins.2014.02.011
0020-0255/

In this paper, we suggest a two-factor face authentication scheme that has an efficient and secure cancellation feature in that the templates composed of permutation and feature vectors can be freely and efficiently changed, thereby resolving the security problems found in the scheme by Kang and colleagues [20]. In Section 2, we briefly look at the security aspects of biometrics, dealing specifically with face authentication. We then explain the basic concept of the scheme in Kang and colleagues and its security problems in Section 3. In Section 4, we introduce our new face authentication system with various considerations pertaining to the scheme. In Section 5, various experimental scenarios and results are presented. Section 6 addresses the security problems of our scheme, which also apply to Kang and colleagues. Section 7 discusses other issues, and Section 8 provides conclusions.

2. Security problems in biometrics

2.1. Biometrics for user authentication

Three methods are used for authenticating an entity in a general system: something you know, something you have, and something you are.

- ***Something you know*** is information possessed, such as passwords or personal identification numbers (PINs).
- ***Something you have*** includes physical items you possess, such as smart cards and universal serial bus (USB) devices.
- ***Something you are*** is bio-information, such as the face, fingerprints, the iris, and the voice. By definition, biometrics refers to methods of verifying the identity of individuals based on physical or behavioral traits, including bio-information, as well as the study of this field.

Bio-information is a strong method when it comes to authentication. For example, passwords (something you know) or hardware tokens (something you have) is easily guessed or stolen, but bio-information (something you are) is not. The probability of finding two persons with identical bio-information is very low. Therefore, it is widely held that bio-information can secure a system against forged authentication, and many organizations that require high security levels have adopted bio-information in their systems. Biometrics has been used in financial services, such as Internet banking and automatic teller machines. Many companies that require a high level of access control have employed biometrics for basic authentication means.

Unlike information utilized in the first two methods, bio-information is very difficult to change. When it comes to security issues, bio-information must not be revealed. If a template (that is, the bio-information stored in the system) is revealed, then it must be removed from the system and replaced with a new template in order to prevent threats to other systems or databases. In addition, bio-information can be physically spoofed [35,5,48].

Biometric data that can be removed from stored templates and placed into new templates are referred to as cancelable biometrics [7,45]. However, this cancel-and-change procedure is difficult to achieve with some biometrics because people cannot offer the same bio-information to the system for every authentication attempt. This uncertainty makes the application of cryptographic one-way functions difficult due to an avalanche effect [56,16].

2.2. Face authentication

In particular, face authentication (the biometrics discussed in this paper) has a slightly different characteristic from other biometrics. Face authentication normally does not require special equipment, such as a fingerprint scanner, a microphone, or an infrared camera. The face provides a natural and initial means of distinguishing people, and thus people do not feel a sense of incongruity about face authentication. Despite those strong points, face authentication has several weak points. Face authentication suffers from relatively low accuracy compared to other systems. Moreover, people do not generally hide their faces, and thus an attacker can easily obtain the face images (s)he wants. Due to these weaknesses, face authentication is often utilized as a supplementary means of authentication.

2.2.1. Linear face recognition and face authentication

When comparing two face images, it cannot be expected that all position will indicate the same features. In linear projection face recognition, in order to allow a certain position to indicate the same feature, a face image is projected to a feature space. This procedure is referred to as "feature extraction." Our scheme utilizes the advantages gained from feature extraction.

Let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ be a set of N training face images. Each face image $\mathbf{x}_i \in \mathbb{R}^n$ can be represented by the feature vector $\mathbf{y}_i \in \mathbb{R}^m$.

$$\mathbf{y}_i = \mathbf{U}\mathbf{x}_i \tag{1}$$

where $\mathbf{U} \in \mathbb{R}^{m \times n}$ is the projection matrix for transforming an n-dimensional image space to an m-dimensional feature space.

Based on the methods for generating a projection matrix $\mathbf{U}$, several methodologies are available for face recognition schemes. For example, principal component analysis (PCA) schemes [3,54] extract a common feature from the complete raw data, and as such can be used to eliminate unessential information from face data. Linear discriminant analysis (LDA, also known as Fisher's linear discriminant) schemes [4,8,29,34,60] maximize the distance between classes and minimize

the distance between images of the same class. LDA-based schemes show high accuracy and can effectively determine which template is closest to a test image. However, many LDA-based schemes suffer from a small sample size, which occurs when the number of samples is smaller than the number of dimensions of the samples.

For a given test face image $\mathbf{x}$, using the same projection matrix $\mathbf{U}$, we can also extract a feature vector $\mathbf{y}$ such that $\mathbf{y} = \mathbf{U}\mathbf{x}$. We can then choose the corresponding face image $\mathbf{x}_j \in \mathbf{X}$ if and only if $\Delta(\mathbf{y}_j, \mathbf{y})$, the distance between $\mathbf{y}_j$ and $\mathbf{y}$, is the closest to zero among all possible distances $\Delta(\mathbf{y}_1, \mathbf{y}), \Delta(\mathbf{y}_2, \mathbf{y}), \ldots, \Delta(\mathbf{y}_N, \mathbf{y})$ and is smaller than a certain threshold θ. Many methods are used for measuring the distance between two vectors. In particular, Euclidean distance and Manhattan distance (also known as city block distance) have been widely used for measuring distance in face recognition studies. The Euclidean distance and the Manhattan distance are known as special cases of the Minkowski distance. The Minkowski distance between two given vectors $\mathbf{q}$ and $\mathbf{r}$ ($\in \mathbb{R}^m$) is defined as

$$\Delta^p_{\mathrm{Mkws}}(\mathbf{q}, \mathbf{r}) = \left(\sum_{a=1}^{m} | \mathbf{q}_{\langle a \rangle} - \mathbf{r}_{\langle a \rangle} |^p \right)^{1/p} \tag{2}$$

where $\mathbf{q}_{\langle a \rangle}$ and $\mathbf{r}_{\langle a \rangle}$ denote the ath element of vectors $\mathbf{q}$ and $\mathbf{r}$. Minkowski distance is Manhattan distance if $p = 1$ and Euclidean distance if $p = 2$. It is well known that the distance-measuring method affects recognition performance [42,11].

Other approaches for face recognition, such as non-linear PCA, self-organizing map (SOM), and neural network (NN), can be found in the literature [1,59,14].

A face authentication system is slightly different from a face recognition system. Both systems commonly require feature extraction and distance measurement. However, a face recognition system is a one-to-many system, whereas a face authentication system is a one-to-one system. A face recognition system searches its template database for the given face, and a face authentication system verifies whether the given face belongs to the given account. Therefore, with the face authentication system, a user must input his or her identifier with his or her face information.

2.2.2. Face recovery attack on a linear projection face authentication system

In a face authentication system, the feature vectors $\mathbf{y}_i$ of training set $\mathbf{X}$ are precomputed and stored in a database. After the pre-computation phase, $\mathbf{X}$ is deleted from the system. However, if $\mathbf{U}$ and $\mathbf{y}_i$ are revealed to an attacker, (s)he can obtain the corresponding face image.

$$\mathbf{x}'_i = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y}_i \simeq \mathbf{U}^+\mathbf{y}_i \tag{3}$$

It should be noted that $\mathbf{U}^T$ is used for converting $\mathbf{U}$ into a square matrix that has an inverse matrix. Of course, since $\mathbf{U}^T\mathbf{U}$ does not hold for all image data, the recovered face image $\mathbf{x}'_i$ is not the same as the original image. If the inverse matrix does not exist, the pseudo-inverse matrix $\mathbf{U}^+$ [36,41] must be used.

In linear projection face authentication systems, unfortunately, $\mathbf{U}$ and $\mathbf{y}_i$ are stored without any conversion. Thus, in linear projection systems, it is necessary to store the projection matrix and feature vectors in the database only after they are securely encoded.

For face authentication, of course, there are other approaches in which the face recovery attack is not directly applicable. Local binary pattern (LBP) [40], for example, extracts feature vectors from face images without the projection matrix. It is well known that LBP is robust against monotonic gray scale transformations, such as illumination, and thus it is widely used for various biometrics, such as fingerprints and the iris [38]. The Gabor filter, which is a linear filter used for edge detection, can also be used for face recognition/authentication.

2.3. Related works

It is noteworthy that binary representation of bio-information is preferred, because many cryptographic functions can easily handle binary representation. By encrypting random bit strings using binary representation, it is possible to realize a secure biometrics system. Nevertheless, it is still difficult to apply cryptographic functions if the number of error bits in the binary representation is larger than a certain threshold. Therefore, in order to construct secure biometrics using binary representation, a high level of accuracy is indispensable.

The relatively low accuracy that originates from the uncertainty of facial features makes it difficult to extract robust binary keys from face images. Owing to this difficulty, many studies of secure face authentication schemes have employed user tokens in their schemes.

Goh and Ngo and their colleagues [13,39] introduced a method for extracting binary keys from face images and user tokens. After extracting a feature vector from a user face, the system quantizes the vector using user tokens. The system then extracts a binary string from the quantized feature vector and a threshold vector. For all elements of the quantized feature vector, if an ith element is larger or equal to the ith element of the threshold vector, it outputs 1; otherwise, it outputs 0. The authors assumed that users never lost, had stolen, shared, or duplicated their user tokens; under this assumption, their scheme showed excellent performance. However, Kong and colleagues [25] performed experiments involving biohashing and showed that biohashing performance was not as high without that assumption. In a receiver operating characteristic (ROC) curve, the biohashing showed worse performance than the base feature extraction scheme. Nanni and Lumini [37]

investigated some empirical tests on biohashing. They also suggested an improved biohashing scheme that consists of several solutions [33] and were able to achieve good performance even with a 'stolen user token' scenario.

Kevenaar and colleagues [21] suggested a fuzzy commitment-based face authentication scheme. Lu and colleagues [31] suggested a similar scheme. The fuzzy commitment proposed by Juels and Watternberg [19] corrects errors from newly input bio-information. Basically, those schemes extract binary keys from face images, and to authenticate users, they compare the binary keys to stored templates. However, binary keys in Lu and colleagues' scheme is too short to use in practice (about 16–36 bits). Feng and colleagues [9,10] suggested a hybrid framework for archiving secure face template protection. In the framework, the random projection, distribution preserving (DP) transform, and cryptographic hash functions are used. The DP transform extracts binary strings from given vectors, as does the biometric discretization process in the original biohashing scheme. The DP transform is carefully designed in order to maximize the distance between classes. Saini and Sinha [46] proposed an optic-based biohashing scheme using joint power spectrum (JPS). Savvides and colleagues [47] proposed cancelable biometric filters based on Kumar and colleagues' correlated face recognition [26]. In the system, encrypted minimum average correlation energy (MACE) filters are employed after training encrypted face images that can be obtained from users' faces and PINs. Kim and Toh [23] proposed a method to enhance face biometric security. They randomized feature vectors using optimized random vectors provided by the system, such that $\mathbf{z} = \mathbf{W} \cdot \mathbf{Ux}$. In a separate paper, they enhanced their method by using a sparse random matrix [24]. They recently illustrated the performance of their method [22]. Sutcu and colleagues [51] proposed a method of extracting keys from given feature vectors. The system generates sketches from given face images and reconstructs templates using the sketches. From the reconstructed templates, the system extracts keys. Lee and colleagues [28] suggested one-time biometric templates for face authentication. When a user provides his/her face with two random matrices $\mathbf{A}_i$ and $\mathbf{b}_i$ for one-time usage, the system permutes and shifts the feature vector $\mathbf{x}$ such that $\mathbf{p}_i = \mathbf{A}_i\mathbf{x} + \mathbf{b}_i$. The system measures the distance between $\mathbf{p}_i$ and the stored template $\mathbf{g}$, which is computed in the same manner; the distance is not affected by $\mathbf{A}_i$ and $\mathbf{b}_i$. If the face authentication successfully finishes, the system updates template $\mathbf{g}$ using the new $\mathbf{A}'$ and $\mathbf{b}'$ such that $\mathbf{A}_{i+1} = \mathbf{A}'\mathbf{A}_t$ and $\mathbf{b}_{t+1} = \mathbf{A}'\mathbf{b}_{i+1} + \mathbf{b}'$.

3. Kang et al. [20]

To achieve a secure face authentication system, many schemes attempt to hide the feature vectors, but not the projection matrix [13,21,39,33,23,24,28,51,31]. Without the projection matrix, $\mathbf{U}$, or the feature vector, $\mathbf{y}$, an attacker might not recover faces. However, the projection matrix contains critical information, and its size is larger than the feature vector. Note that if the projection matrix is made with PCA, the value of the first element of a feature vector becomes dominant. By searching only a thousand cases, an attacker is able to obtain many human-like face images.

Kang and colleages in 2005 suggested a two-factor face authentication scheme, in contrast with other schemes, that hides the projection matrix [20]. By employing passwords that are decided by users, the face authentication system permutes a projection matrix $\mathbf{U}$ and stores passwords with weight factors $\{\mathbf{y}_1, \mathbf{y}_2, \ldots\}$ as face templates without the user passwords in the database. When a user inputs a password π_i and a face image $\mathbf{x}$ with his/her identifier i, the face authentication system permutes $\mathbf{x}$ using π_i. The system then projects the permuted face image into the permuted projection matrix such that

$$\mathbf{y} = \mathbf{Ux} = f_{c,\pi_i}(\mathbf{U}) f_{r,\pi_i}(\mathbf{x}) \tag{4}$$

where $f_{c,\pi_i}()$ and $f_{r,\pi_i}()$ denote the permutation functions that permute a matrix or vector by column or row according to the password π_i. Corresponding to the identifier i, the face authentication system then tries to match the feature vector $\mathbf{y}$ to the templates that are stored in the database. The system needs to store several projection matrices $\{f_{c,\pi_i}(\mathbf{U})\}_{i\in[1\ldots c]}$ instead of a global projection matrix $\mathbf{U}$, where c is the number of users. Also, when it is necessary to change the stored face templates, this can be achieved by simply changing the password. Therefore, Kang's scheme makes it difficult for an attacker who successfully steals face templates to obtain original face images, and allows users to securely change their old face templates even when their faces have been disclosed.

However, this scheme still suffers from several problems, which are discussed below.

3.1. Exploiting attack

In the scheme by Kang and colleagues [20], permuted face images hold mathematical distances similar to those of the original face images. In such cases, if the user has a face with a distance distinguishable from other users, that user becomes a good target. With a known face image, even an attacker with the corresponding password can check whether the corresponding user is registered in the system.

As shown in Fig. 1, face images recovered via the method delineated by Eq. (3) hold similar mathematical distances if they belong to the same class. The same problem arises for Kim and Toh [23] and Lee and colleagues [28].

3.2. Dictionary attack

In general, usage of simple passwords weakens the system against dictionary attacks. If an attacker obtains plain data and the encrypted form of the data, (s)he can attempt to find the encryption key by performing encryption of the plain data with all the possible keys in the prepared dictionary.

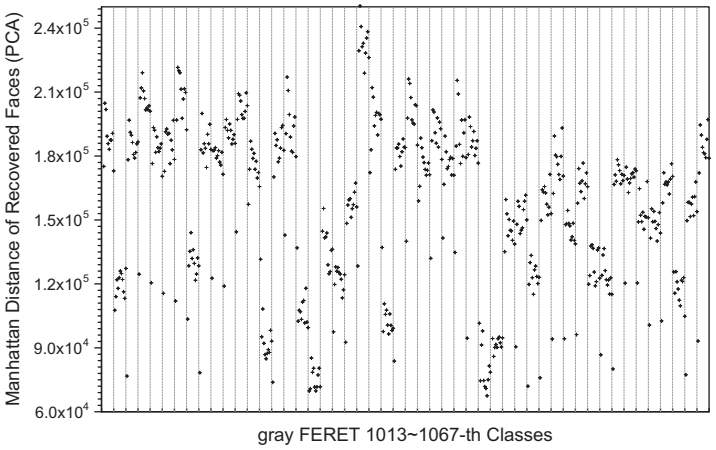


Fig. 1. Distribution of distances of recovered face images (PCA, gray FERET 1013–1067th classes).

In the scheme by Kang and colleagues [20], by a single dictionary attack on a stolen template, an attacker can successfully obtain the projection matrix **U**. After obtaining the projection matrix, the attacker can recover all face images in the database, as delineated in Eq. (3). This means that the use of passwords is not desirable in their scheme.

4. New face authentication scheme

In this section, we describe a new face authentication scheme that has an efficient and secure cancellation feature, in that the templates composed of permutation and feature vectors can be freely and efficiently changed, thereby resolving the security problems seen in Kang and colleagues [20]. Our scheme is similar to linear face authentication systems, where feature vectors are expressed in Eq. (1). As shown in Section 2.2.2, discovery of the feature vectors $\mathbf{y}_i$ and the projection matrix **U** allows an attacker to rebuild faces. Thus, we hide that information by saving **Sy** and $\mathbf{SUP}^{-1}$ instead of **y** and **U**, where $\mathbf{P} \in \mathbb{R}^{n\times n}$ is a permutation matrix, and $\mathbf{S} \in \mathbb{R}^{m\times m}$ is an isolation matrix. The isolation matrix **S** prevents an attacker who succeeds in breaking into an account from successfully attacking other accounts. In Section 4.2, we show that faces can be matched with **Sy** and $\mathbf{SUP}^{-1}$ instead of **y** and **U**.

4.1. System setup and user enrollment

Let $\mathbf{U} \in \mathbb{R}^{m\times n}$ be a projection matrix that is obtained from training sets $\{\mathbf{X}_i\}_{i\in[1\ldots c]}$ using conventional feature extraction schemes such as eigenface [3,54] or Fisherface [4,29,32,34,60], where $\mathbf{X}_i = \{\mathbf{x}_{(i,1)}, \mathbf{x}_{(i,2)}, \ldots \in \mathbb{R}^n\}$ denotes a set of face images of the ith user, and c is the number of classes. Also, let $\mathbf{Y}_i = \{\mathbf{y}_{(i,1)}, \mathbf{y}_{(i,2)}, \ldots \in \mathbb{R}^m\}$ be a set of feature vectors corresponding to the training set $\mathbf{X}_i$ such that $\mathbf{y}_{(i,j)} = \mathbf{U}\mathbf{x}_{(i,j)}$.

After obtaining **U** and $\{\mathbf{Y}_i\}_{i\in[1\ldots c]}$, our new face authentication system generates random isolation matrices $\{\mathbf{S}_i \in \mathbb{R}^{m\times m}\}_{i\in[1\ldots c]}$ for all users. Every **S** must satisfy

$$\sum_{a=1}^{m}\mathbf{S}_{\langle a,b\rangle} = 1, \quad \sum_{b=1}^{m}\mathbf{S}_{\langle a,b\rangle} = 1, \quad \mathbf{S}_{\langle a,b\rangle} \in \{0,1\} \tag{5}$$

for $1 \leqslant a, b \leqslant m$, where $\mathbf{S}_{\langle a,b\rangle}$ denotes an element of **S** in the ath row and the bth column. Optionally, the system can use one isolation matrix **S** for all users. The system can generate **S** as described in Algorithm 1.

The face authentication system then receives user passwords $\{\pi_i\}_{i\in[1\ldots c]}$ from all users and randomly chooses salts $\{\kappa_i\}_{i\in[1\ldots c]}$. The system computes the inverse permutation matrices $\{\mathbf{P}_i^{-1}\}_{i\in[1\ldots c]}$ using these passwords and salts, as described in Algorithm 2. Algorithm 2 receives kd as an input, where kd is given to adjust the degree of influence of the user password on face authentication. When kd is small, the influence of the user password on face authentication is small and vice versa. This aspect is demonstrated in the results of our experiments. Please note that if $\mathrm{kd} = 0$, the permutation matrices in our scheme are the same as those for Kang and colleagues [20].

Unlike conventional systems, which store $\langle \mathbf{Y}_i, \mathbf{U}\rangle$ as a face template for the ith user, a system that adopts our face authentication scheme stores $\mathcal{T}_i = \langle \kappa_i, \mathcal{W}_i, \mathbf{Q}_i\rangle$, where κ_i is a salt, $\mathcal{W}_i = \mathbf{S}_i\mathbf{Y}_i = \{\mathbf{S}_i\mathbf{y}_{(i,1)}, \mathbf{S}_i\mathbf{y}_{(i,2)}, \ldots\}$, and $\mathbf{Q}_i = \mathbf{S}_i\mathbf{U}\mathbf{P}_i^{-1}$. After storing all templates, the system should delete all other information such as $\mathbf{U}, \{\mathbf{S}_i, \pi_i, \mathbf{P}_i, \mathbf{P}_i^{-1}, \mathbf{X}_i, \mathbf{Y}_i\}_{i\in[1\ldots c]}$.

Normally, **U** is shared for all users, and hence the conventional system stores only one **U**. In our system, however, all the different values of $\mathbf{Q}_i$ must be stored. This requires much more storage space, but the problem of storage capacity is negligible in most face authentication systems today.

Algorithm 1. Generating isolation matrix **S**

Require: r
seed_rand(r)
$\mathbf{S} \leftarrow \mathbf{I} \in \mathbb{R}^{m \times m}$ /* identity matrix */

for $a = 1$ **to** m **do**
  $c \leftarrow \mathrm{int}(\mathrm{rand}(1, m))$
  for $b = 1$ **to** m **do**
    $\mathrm{swap}(\mathbf{S}\langle a, b\rangle, \mathbf{S}\langle c, b\rangle)$
  end for
end for

return S

Algorithm 2. Generating permutation matrix **P**

Require: π, κ, kd, ...
seed_rand(π, κ, ...)
$\mathbf{P} \leftarrow \mathbf{I} \in \mathbb{R}^{n \times n}$ /* identity matrix */

for $a = 1$ **to** n **do**
  $\mathbf{P}_{\langle (a,a)\rangle} \leftarrow \mathrm{rand}(l - \mathrm{kd}, 1 + \mathrm{kd})$
end for

for $a = 1$ **to** n **do**
  $c \rightarrow \mathrm{int}(\mathrm{rand}(1, n))$
  for $b = 1$ **to** n **do**
    $\mathrm{swap}(\mathbf{P}_{\langle a,b\rangle}, \mathbf{P}_{\langle c,b\rangle})$
  end for
end for

return P

4.2. Face authentication

Now, a user whose account number is i inputs his/her face $\mathbf{x} \in \mathbb{R}^n$ and password π with his/her ID into the face authentication system. The system fetches the corresponding user template $\mathcal{T}_i = \langle \kappa_i, \mathcal{W}_i, \mathbf{Q}_i \rangle$ from the database. Using π and κ_i, the system generates the permutation matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$ according to Algorithm 2 and computes the permuted face $\mathbf{Px} \in \mathbb{R}^{n \times n}$. With the permuted face **Px**, the system computes $\mathbf{t} = \mathbf{Q}_i \cdot \mathbf{Px} \in \mathbb{R}^m$. The system measures the distances between **t** and one element of $\mathcal{W}_i = \{\mathbf{S}_i\mathbf{y}_{(i,1)}, \mathbf{S}_i\mathbf{y}_{(i,2)}, \ldots \in \mathbb{R}^m\}$. If the distances are shorter than the threshold value, then the system grants the user access to the system. Otherwise, the system denies access.

If the user correctly inputs **x**, π, and the ID (i.e., i), then $\mathbf{P}_i^{-1} \cdot \mathbf{P} = \mathbf{I} \in \mathbb{R}^{n \times n}$, which is the $n \times n$ identity matrix. Thus, **t** should be equal to $\mathbf{S}_i\mathbf{Ux} = \mathbf{S}_i\mathbf{y}$. The distance between **t** and one element of $\mathcal{W}_i$, $\Delta(\mathbf{t}, \mathbf{S}_i\mathbf{y}_{(i,j)}) = \Delta(\mathbf{S}_i\mathbf{y}, \mathbf{S}_i\mathbf{y}_{(i,j)})$ is equal to $\Delta(\mathbf{y}, \mathbf{y}_{(i,j)})$ because

$$\Delta^p_{\mathrm{Mkws}}(\mathbf{S}_i\mathbf{y}, \mathbf{S}_i\mathbf{y}_{(i,j)}) = \left(\sum_{a=1}^{m} \left| (\mathbf{S}_i\mathbf{y})_{\langle a\rangle} - (\mathbf{S}_i\mathbf{y}_{(i,j)})_{\langle a\rangle} \right|^p\right)^{1/p} = \left(\sum_{a=1}^{m} \left| \sum_{b=1}^{m} (\mathbf{S}_i)_{\langle a,b\rangle}\mathbf{y}_{\langle b\rangle} - \sum_{b=1}^{m} (\mathbf{S}_i)_{\langle a,b\rangle}(\mathbf{y}_{(i,j)})_{\langle b\rangle} \right|^p\right)^{1/p}$$
$$= \left(\sum_{a=1}^{m} \left| \mathbf{y}_{\langle \epsilon_a\rangle} - (\mathbf{y}_{i,j})_{\langle \epsilon_a\rangle} \right|^p\right)^{1/p} = \Delta^p_{\mathrm{Mkws}}(\mathbf{y}, \mathbf{y}_{(i,j)}) \tag{6}$$

where ϵ_a denotes an index in the ath row of $\mathbf{S}$; $\mathbf{S}_{\langle a,\epsilon_a\rangle} = 1$ and $\mathbf{S}_{\langle a,j\rangle} = 0$ for $j \neq \epsilon_a$. Note that $\epsilon_1 \neq \epsilon_2 \neq \cdots \neq \epsilon_m$ and $\epsilon_a \in [1 \ldots m]$. Thus, the result of the distance measurement in our system is the same as a conventional face authentication system.

A system that adopts our face authentication scheme is illustrated in Fig. 2. The system consists of five parts: acquisition, account management, face permutation, a database, and a matching routine. Each routine can be placed in a physically distant location.

- ***Acquisition***: When a user wants to be authenticated in a system that adopts our scheme, (s)he offers his/her face image $\mathbf{x}$, password π, and identifier ID to the acquisition routine. The acquisition routine consists of an optical device for obtaining the user's facial image and a key pad for the password. After receiving the facial image and password, the acquisition routine passes them to the face permutation routine. The ID is passed to the account management routine.
- ***Account management***: The account management routine finds the user's account number i using the user identifier ID, which was handed over from the acquisition routine, and sends it to the database.
- ***Face permutation***: When the face permutation routine receives the user's facial image $\mathbf{x}$, password π, and salt κ_i, it generates a permutation matrix $\mathbf{P}$ and computes the permuted face image $\mathbf{Px}$. After computing $\mathbf{Px}$, the face image permutation routine sends it to the matching routine.
- ***Database***: According to the user account number i, the database sends the salt κ_i to the face permutation routine and sends the permuted projection matrix $\mathbf{Q}_i = \mathbf{S}_i\mathbf{U}\mathbf{P}^{-1}$ and the permuted feature vectors $\mathcal{W}_i = \{\mathbf{S}_i\mathbf{y}_{(i,j)}\}_j$ to the matching routine.
- ***Matching***: Using the permuted face image $\mathbf{Px}$ and projection matrix $\mathbf{Q}_i$ received from the face permutation routine and the database, respectively, the matching routine computes the permuted feature vector $\mathbf{t} = \mathbf{Q}_i \cdot \mathbf{Px}$. After computing $\Delta(\mathcal{W}_i, \mathbf{t})$, the distances between the feature vector $\mathbf{t}$ and all the stored feature vectors $\mathcal{W}_i = \{\mathbf{S}_i\mathbf{y}_{(i,j)}\}_j$, the matching routine returns 1 when the distances are lower than a certain threshold value. Otherwise, the matching routine returns 0.

4.3. Cancellation of templates

The template used in our face authentication scheme is defined as $\mathcal{T}_i = \langle \kappa_i, \mathcal{W}_i, \mathbf{Q}_i\rangle$. Let $\mathcal{T}_{i,\text{old}} = \langle \kappa_{i,\text{old}}, \mathcal{W}_{i,\text{old}}, \mathbf{Q}_{i,\text{old}}\rangle$ be an old template of a user i. When the user wishes to change his/her template due to security risks or requests, the system cancels the old template and replaces it with a new one. To perform this operation, the system must know $\mathbf{P}_{i,\text{old}}\mathbf{P}^{-1}_{i,\text{new}}$; it can then compute $\mathbf{S}_{i,\text{old}}\mathbf{U}\mathbf{P}^{-1}_{i,\text{new}}$.

$$\mathbf{Q}_{i,\text{old}} \cdot \mathbf{P}_{i,\text{old}}\mathbf{P}^{-1}_{i,\text{new}} = \mathbf{S}_{i,\text{old}}\mathbf{U}\mathbf{P}^{-1}_{i,\text{old}} \cdot \mathbf{P}_{i,\text{old}}\mathbf{P}^{-1}_{i,\text{new}} = \mathbf{S}_{i,\text{old}}\mathbf{U}\mathbf{P}^{-1}_{i,\text{new}} \tag{7}$$

To support this operation, the user simply offers the old password π_{old} and a new password π_{new} in sequence. The system chooses a new salt $\kappa_{i,\text{new}}$. From $(\pi_{\text{old}}, \kappa_{i,\text{old}})$ and $(\pi_{\text{new}}, \kappa_{i,\text{new}})$, the system can compute $\mathbf{P}_{i,\text{old}}$ and $\mathbf{P}_{i,\text{new}}$.

However, an attacker who succeeds in breaching the database can still ascertain which template belongs to which user by observing $\mathcal{W}_{i,\text{old}}$. To break the relationship between the old and the new templates, the system must also change the isolation matrix $\mathbf{S}_{i,\text{old}}$ by multiplying a new isolation matrix $\mathbf{S}_{i,\text{add}}$.

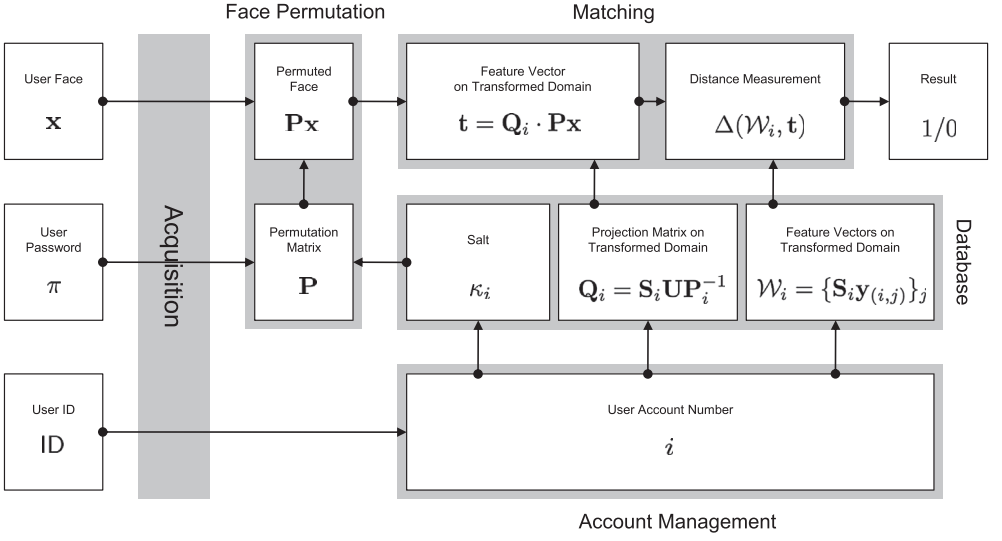


Fig. 2. System illustration of a system that adopts our face authentication scheme; it consists of acquisition, account management, face permutation, a database, and matching.

$$\mathcal{W}_{i,\text{new}} = \{\mathbf{S}_{i,\text{add}} \cdot \mathbf{S}_{i,\text{old}}\mathbf{y}_{(i,j)}\}_j \tag{8}$$

$$\mathbf{Q}_{i,\text{new}} = \mathbf{S}_{i,\text{add}} \cdot (\mathbf{Q}_{i,\text{old}} \cdot \mathbf{P}_{i,\text{old}}\mathbf{P}_{i,\text{new}}^{-1}) = \mathbf{S}_{i,\text{add}} \cdot \mathbf{S}_{i,\text{old}}\mathbf{U}\mathbf{P}_{i,\text{new}}^{-1} = \mathbf{S}_{i,\text{new}}\mathbf{U}\mathbf{P}_{i,\text{new}}^{-1} \tag{9}$$

The system then deletes $\mathcal{T}_{i,\text{old}}$ from the database and stores $\mathcal{T}_{i,\text{new}} = \langle \kappa_{i,\text{new}}, \mathcal{W}_{i,\text{new}}, \mathbf{Q}_{i,\text{new}} \rangle$ as a new template. The new isolation matrix $\mathbf{S}_{i,\text{add}}$ must also be deleted from the system.

4.4. Permutation matrix and isolation matrix

4.4.1. User password

An attacker who succeeds in invading the system might try to recover a certain user's face. However, the spaces of $\mathbf{y}$ and $\mathbf{U}$ are too large to decompose $\mathbf{S}_i\mathbf{y}$ and $\mathbf{Q}_i(=\mathbf{S}_i\mathbf{U}\mathbf{P}_i^{-1})$ and to determine unique values of $\mathbf{y}$ and $\mathbf{U}$. To decompose them, an attacker must search $m!/2$ possible cases to find out $\mathbf{S} \in \mathbb{R}^{m\times m}$ on average, because each row and column of $\mathbf{S}$ has only one 1, and other cells are all 0. For example, for $m = 50, m!/2 \approx 1.52 \times 10^{64} > 2^{213}$. An exhaustive search for this space is not feasible. Without $\mathbf{y}_i$ and $\mathbf{U}$, the attacker cannot recover a user face image $\mathbf{x}'$ but rather can only obtain a permuted face image $\mathbf{P}_i\mathbf{x}'$ as follows:

$$\mathbf{P}_i\mathbf{x}' = (\mathbf{Q}_i^T\mathbf{Q}_i)^{-1}\mathbf{Q}_i^T\mathbf{S}\mathbf{y} \simeq \mathbf{Q}_i^+\mathbf{S}\mathbf{y}. \tag{10}$$

A permutation matrix $\mathbf{P}_i$ prevents an attacker who recovered a permuted face image $\mathbf{P}_i\mathbf{x}'$ from obtaining the face image $\mathbf{x}'$. In general, several sources can be used in generating a permutation matrix $\mathbf{P}_i$, such as a USB token, smart card, or password. Each source has its respective advantages and disadvantages. For example, the USB token provides high entropy to the permutation matrix, but a user must carry the device. The password, which is "something you know," offers a user convenience, but the entropy of the password is not sufficiently high. That is, if a password is adopted in making the permutation matrix $\mathbf{P}_i$, an attacker who has recovered a permuted face image $\mathbf{P}_i\mathbf{x}'$ can additionally attempt a dictionary attack to remove $\mathbf{P}_i$ from $\mathbf{P}_i\mathbf{x}'$. Therefore, for secure face authentication, the system should not use a password for generating permutation matrices. In some systems, however, the use of a password, despite its low entropy, is unavoidable, and thus the password option should be supported in our system as a possible input source.

Our scheme resembles the biohashing scheme [13,39] in that both schemes involve some amount of user-supplied information. The user token in the biohashing scheme quantizes the projected face image (not the face image itself), whereas the user password in our scheme permutes the face image. Since the quantization in biohashing uses a user token, face authentication is directly affected by this, and the overall performance of biohashing thereby increases. In other words, the user token compensates for the information loss during dimensional reduction. The user password in our scheme also affects the face authentication, as it changes the distance between the feature vector and templates when different classes are met, as shown in the experiment results in Section 5. However, the user password does not shorten the distance within the same classes. Thus, the password in our scheme does not compensate for any information in face authentication.

4.4.2. Salt

The salt κ_i is a random bit string for user i that is used to prevent complicated dictionary attacks that use pre-computation of dictionary words. Since a permutation matrix $\mathbf{P}$ is generated from user passwords, in order to prevent a pre-computed dictionary attack, the salt is essential in our scheme. In addition, if a permutation matrix $\mathbf{P}$ does not have an inverse matrix, it is necessary to replace the salt κ with a different value.

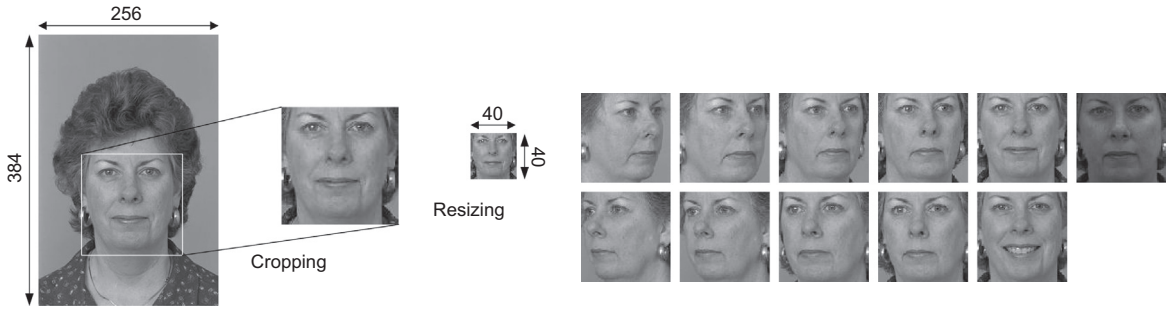
4.4.3. Isolation matrix

When a system uses a password, an isolation matrix $\mathbf{S}_i$ prevents an attacker who has succeeded in recovering a face image $\mathbf{x}'$ from recovering face images of other users with only a single dictionary attack. The attacker can remove $\mathbf{P}^{-1}$ from $\mathbf{Q}_i = \mathbf{S}_i\mathbf{U}\mathbf{P}^{-1}$ if (s)he succeeds in the dictionary attack, but (s)he cannot obtain $\mathbf{U}$ without removing $\mathbf{S}_i$ from $\mathbf{S}_i\mathbf{U}$. Therefore, to recover face images of other users, the attacker must independently perform multiple dictionary attacks. Similar to permutation matrices, isolation matrices are easily generated from random bit strings, as shown in Algorithm 1.

We assume that each user has his/her own isolation matrix, but even a single shared isolation matrix guarantees sufficient security of the projection matrix $\mathbf{U}$ and feature vectors $\{\mathbf{Y}_i\}_{i\in[1\ldots c]}$. An exhaustive search for the single isolation matrix is not feasible in polynomial time, as we discuss in Section 4.4.1. Therefore, the system can optionally use an isolation matrix for all users.

The isolation matrices must be deleted from the system after user templates are set up. Note that a matrix that satisfies Eq. (5) is orthogonal. Orthogonal matrices guarantee the equality of the Euclidean distance between two vectors. For example, given an orthogonal matrix $\mathbf{A}$ ($\mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A} = \mathbf{I}$), $\Delta_{\text{Euc}}(\mathbf{A}\mathbf{q}, \mathbf{A}\mathbf{r}) = \Delta_{\text{Euc}}(\mathbf{q}, \mathbf{r})$. However, some orthogonal matrices do not guarantee the equality of the Manhattan distance, and therefore the isolation matrices must satisfy Eq. (5).

Please note that isolation matrices do not affect the results of face authentication at all.



(a) Face cropping and resizing

(b) Cropped faces from facial recognition technology database (1013-th class)

Fig. 3. Samples of face images for the training and testing set.

4.5. Security enhancement

4.5.1. Usage of other secure information

To guarantee sufficient security, the system must ensure users employ high-entropy passwords, because user passwords are weak against dictionary attacks. However, other external security information can be used for our scheme. As noted earlier, smart cards and USB devices are known to be more secure than user passwords. These secure means can be used in authentication procedures in place of bio-information due to their high security features. However, it must be noted that they are only an indirect means against authentication forgery.

4.5.2. Dummy templates

By using linear projection methods in face authentication, it is difficult to process a number of classes due to the size of the projection matrix and the capacity of classifiers. Moreover, with a small number of classes, an attacker can easily trace the canceled templates with high probability. With many dummy templates, the attacker cannot easily determine the raw projection matrix and has to expend more time and effort to trace the canceled templates. Since these dummy templates, based on real faces, affect face authentication, the dummy templates need to be randomly generated, but should be out of range of the real faces.

5. Experiments and discussions

In our experiments, we used 194 classes (1013–1206th classes) from the gray facial recognition technology (FERET) database [43,44]. Each class has 11 different face images. Among the 11 face images, 10 random images were used for training, and the remaining image was used for testing. Each image was manually cropped to the face only and resized to 40×40 pixels, as shown in Fig. 3. Except for cropping and resizing, no other pre-processing was applied.

We used PCA and LDA for the feature extraction scheme. The size of the projection matrix was 50 for PCA (i.e., $\mathbf{U}_{\mathrm{PCA}} \in \mathbb{R}^{50\times 1600}$). Regularized discriminant analysis (R-LDA) [32] was used for representation of the numerous LDA extensions, dealing with the small sample size problem, such as null space LDA (N-LDA) [8,30] and uncorrelated LDA (U-LDA) [17]. R-LDA generates projection matrices that are automatically adjusted from 42 to 154 by the scheme itself, according to the size of the training set (i.e., $\mathbf{U}_{\mathrm{LDA}} \in \mathbb{R}^{42\times 1600} \sim \mathbb{R}^{154\times 1600}$). In addition, we used the Manhattan distance when distance measurement $\Delta(\cdot,\cdot)$ was required.

For the experiments, we used Mathwork MATLAB (ver. R2009a, R2012b) on a personal computer with an Intel Core2Duo E6600 processor and 8 GB RAM running under Windows 7.[1] Table 1 shows time consumption for major operations.

Hereafter, for simplicity of description, we use the terms S-PT (simple permutation transformation) for the scheme in Kang and colleagues [20] and G-PT (general permutation transformation) for our new scheme. Using these terms, denotations for the base face authentication methods were established. For example, S-PT LDA denotes the scheme in Kang and colleagues based on LDA, and G-PT PCA refers to our new method based on PCA.

5.1. Face authentication

- *Scenario* 1
  Users who were registered in the face authentication system attempt to pass the authentication process by offering their IDs, passwords, and face images.

[1] The MATLAB source code is available at http://ex.isrl.kr/isrl-bio-gpt-code.zip.

Table 1
Time consumption for major operations.

The number of classes		54	194
Generation projection matrix	PCA: $\{\mathbf{X}_i\}_{i\in[1\ldots c]} \rightarrow \mathbf{U}_{\text{PCA}}$	157.2 ± 30.9	320.8 ± 45.1
	LDA: $\{\mathbf{X}_i\}_{i\in[1\ldots c]} \rightarrow \mathbf{U}_{\text{LDA}}$	1.5 ± 0.3	294.8 ± 0.6
Generating isolation matrix	$\mathbf{S}$	0.144 ± 0.292	
Generating permutation matrix	S-PT: $(\pi, \kappa) \rightarrow \mathbf{P}$	43.3 ± 3.0	
	G-PT: $(\pi, \kappa, \text{kd}) \rightarrow \mathbf{P}$	190.1 ± 3.6	
Generating permuted face	S-PT: $\mathbf{Px}$	4.3 ± 1.7	
	G-PT: $\mathbf{Px}$	5.1 ± 2.6	
Projecting (Feature extracting)	$\mathbf{t} = \mathbf{Q} \cdot \mathbf{Px}$	0.158 ± 0.364	
Transposing/Inversing matrix	S-PT: $\mathbf{P} \rightarrow \mathbf{P}^T$	11.8 ± 0.2	
	G-PT: $\mathbf{P} \rightarrow \mathbf{P}^{-1}$	366.0 ± 42.3	
Measuring distance (Matching)	$\Delta^1_{\text{Mkws}}(\mathcal{W}, \mathbf{t}) = \Delta_{\text{Man}}(\mathcal{W}, \mathbf{t})$	0.128 ± 0.296	
	$\Delta^2_{\text{Mkws}}(\mathcal{W}, \mathbf{t}) = \Delta_{\text{Euc}}(\mathcal{W}, \mathbf{t})$	0.129 ± 0.265	

(Milliseconds, Intel Core2Duo E6600, MATLAB R2012b.)

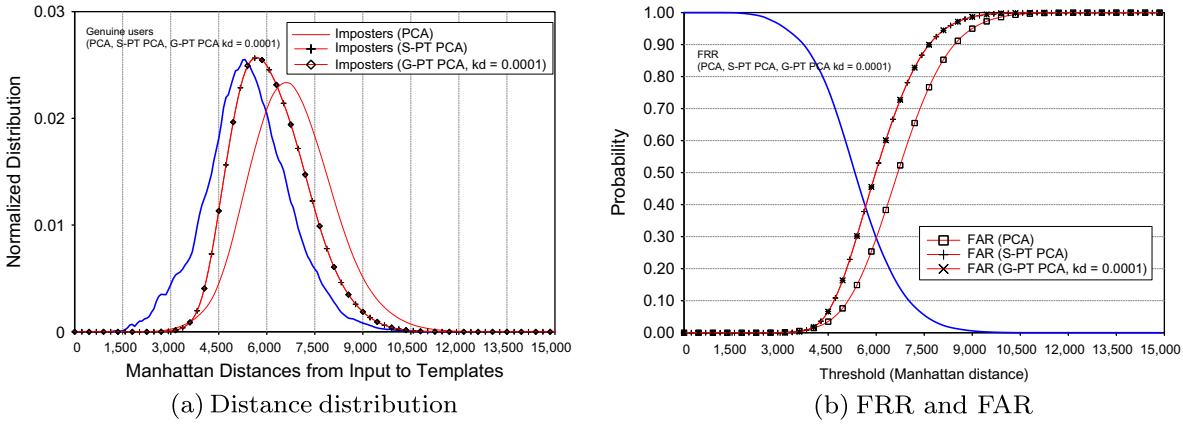


Fig. 4. Comparison between PCA, S-PT PCA, and G-PT PCA (number of classes = 194, kd = 0.0001).

- *Experiment* 1
  We performed face authentication experiments using $54, 74, \ldots, 194$ classes. For each class, only one face image, randomly selected from among 11 face images, was used for testing. We repeated each experiment 100 times for LDA/PCA, S-PT LDA/PCA, and G-PT LDA/PCA. To measure the influence of the user password, we performed additional experiments for G-PT LDA with different kd values.
- *Result* 1
  For a face image with its class identifier, face authentication checks whether the face indeed belongs to the corresponding class. It seems that different classes are not considered in face authentication. However, the distance between classes affects the determination of threshold values. These values change the false acceptance ratio (FAR), and thus they also affect the equal error ratio (EER), but not the false rejection ratio (FRR).

Fig. 4 shows the authentication performance of S-PT/G-PT PCA (small kd). In Fig. 4(a), the x-axis denotes the discrete distances between the feature vector and stored templates, and the y-axis denotes the normalized distribution, which is the probability of occurrences in each distance interval. S-PT and G-PT PCA (small kd) show worse authentication performance than PCA; imposters are not distinguishable from genuine users. As shown in Fig. 4(a), the distance between feature vectors of imposters and templates in S-PT/G-PT PCA becomes smaller. EER from PCA to S-PT PCA increases from 29.6% to 39.3%. For certain thresholds, FAR becomes higher, as shown in Fig. 4(b). This situation means that more imposters can pass authentication process, or more genuine users cannot pass the authentication, if S-PT or G-PT PCA (small kd) is used. This situation is undesirable in a face authentication system.

We infer that these results are due to the characteristics of PCA. PCA extracts the common feature (i.e., eigenvectors) from given training data, and thus the feature vectors of every class are closely located in a small vector space. Basically, our scheme relocates feature vectors of imposters' faces. This means that our scheme with a small kd relocates the feature vectors in a smaller vector space than PCA.

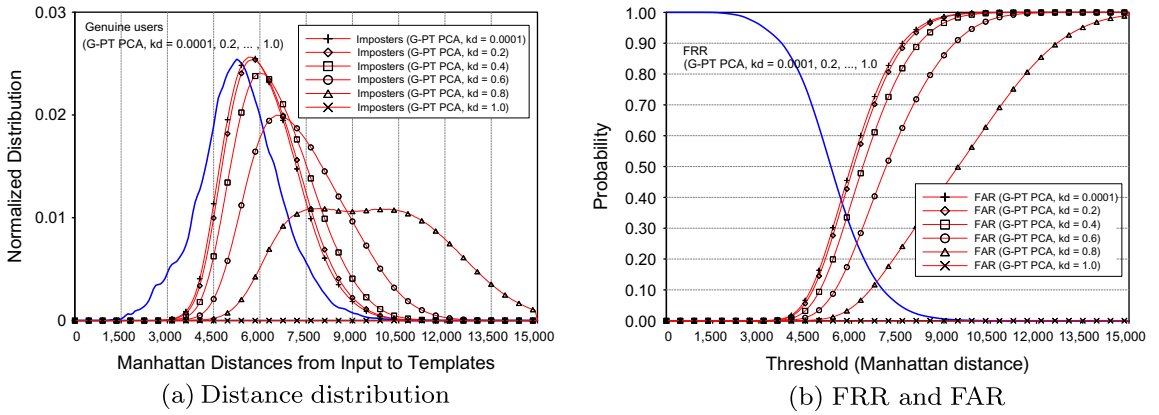


Fig. 5. Comparison between G-PT PCAs with different kd values (number of classes = 194).

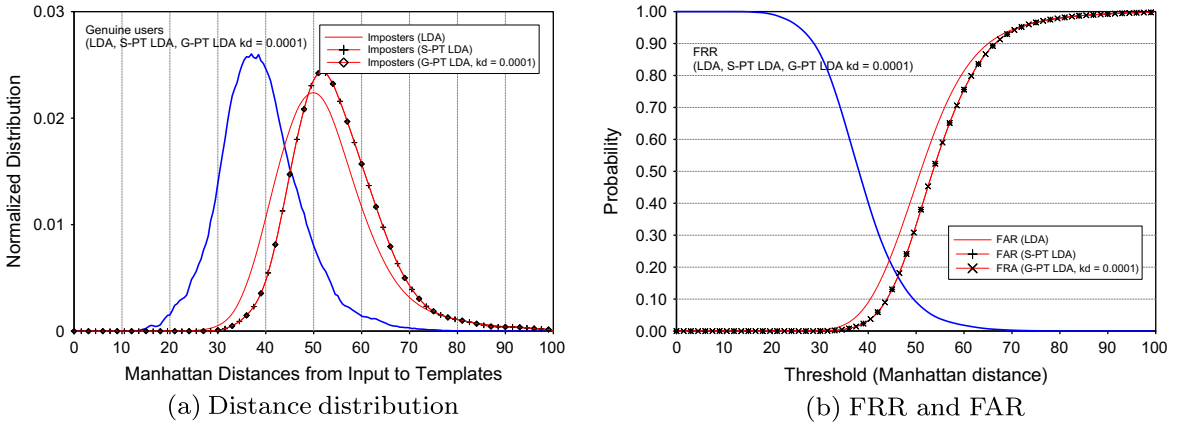


Fig. 6. Comparison between LDA, S-PT LDA, and G-PT LDA (number of classes = 194, kd = 0.0001).

Fig. 5 shows the performance of G-PT PCAs with different kd values. As kd increased, the average distance between feature vectors of imposters and the templates grew. When kd = 1.0, EER approaches zero (0.002–0.003%) and an FRR of zero FAR is 23.9%. The unusual FRR is due to one case among 37,442,000 trials, except that FRR is reduced to 11.5%.

S-PT/G-PT LDA (small kd) shows better authentication performance than the original LDA. As shown in Fig. 6(a), the distance between feature vectors of imposters and the templates in S-PT/G-PT LDA became greater. This increase directly affects the FAR, and it also reduces EER from 22.2% to 17.1%, as shown in Fig. 6(b). However, an EER of 17.1%, which is the limitation of S-PT, is still not applicable in the real world.

Fig. 7 shows the performance of G-PT LDA's with different kd values. As kd increased, the average distance between feature vectors of imposters and the templates grew. Specifically, when kd = 1.0, both FAR and EER become zero. This means that no one can pass authentication without a correct password. This is very desirable in a secure authentication system.

Table 2 shows FRR of zero FAR and EER in detail.

- *Scenario* 2
  Sometimes genuine users forget their passwords. In this case, they are referred to as "password imposters."
- *Experiment* 2
  For 194 classes, we performed experiments on G-PT PCA and G-PT LDA with different kd values. Instead of using correct passwords for constructing templates, we input random passwords. We repeated each experiment 100 times.
- *Result* 2
  Fig. 8 shows the distance distribution and FAR of password imposters. When different passwords were used, the face authentication system computed $\mathbf{t} = \mathbf{Q}_i \cdot \mathbf{P}_j\mathbf{x} = \mathbf{S}_i\mathbf{U}\mathbf{P}_i^{-1} \cdot \mathbf{P}_j\mathbf{x}$ for $i \neq j$. In general, $\mathbf{P}_i^{-1} \cdot \mathbf{P}_j$ is not equal to $\mathbf{I}$. This is similar to a situation where general imposters input wrong faces and passwords. Thus, we can expect Fig. 8 to resemble Figs. 5 and 7. However, the performance in this case is slightly lower than in Scenario 1.

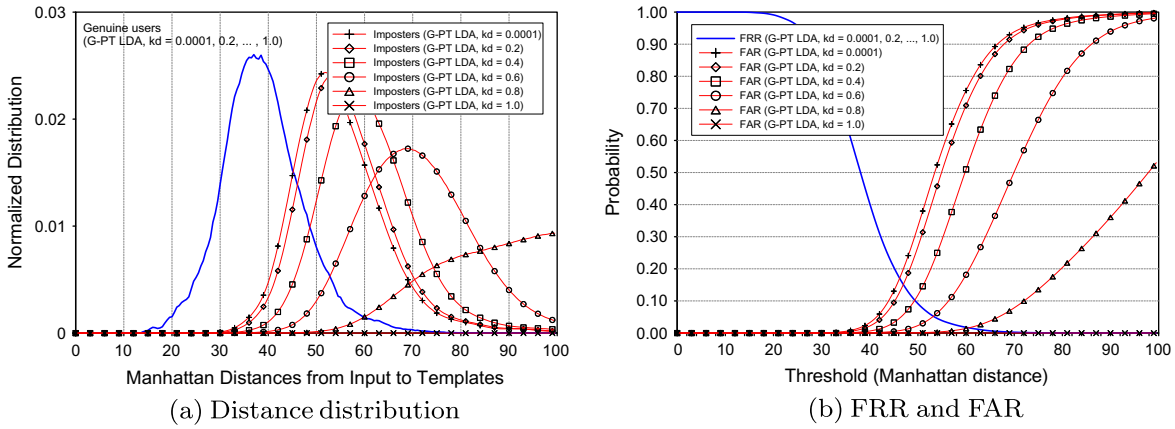


Fig. 7. Comparison between G-PT LDA's with different kd values (number of classes = 194).

Table 2
FRR of Zero FAR and the EER of Scenario 1 (%).

Numbers of classes	FRR of zero FAR								EER							
	54	74	94	114	134	154	174	194	54	74	94	114	134	154	174	194
PCA	97.4	97.7	97.7	97.7	97.3	97.3	97.3	97.3	30.0	29.8	29.7	29.7	29.6	29.7	29.6	29.6
S-PT PCA	95.3	95.6	95.6	95.6	95.6	95.1	95.1	95.1	39.3	39.3	39.2	39.3	39.3	39.3	39.2	39.3
G-PT PCA																
kd = 0.0001	95.6	95.6	95.7	95.5	95.7	95.6	95.1	95.2	39.3	39.4	38.7	39.5	39.5	39.4	39.3	39.3
kd = 0.2	95.4	95.2	95.2	95.1	95.2	95.0	95.2	95.1	37.7	37.9	38.1	38.3	38.1	38.1	38.0	38.0
kd = 0.4	94.2	94.1	93.8	93.9	94.1	94.0	94.0	94.0	34.0	34.0	33.3	33.5	33.8	33.7	33.9	33.6
kd = 0.6	91.3	91.6	91.3	91.4	91.3	91.4	91.3	91.4	26.0	25.3	25.2	25.4	25.4	25.5	25.3	25.1
kd = 0.8	80.5	80.0	81.8	80.5	81.1	82.5	81.1	81.2	12.9	12.7	12.7	12.7	12.9	13.0	12.7	12.6
kd = 1.0	2.1	4.7	17.2	6.1	24.0	55.6	6.2	23.9	0.002	0.003	0.002	0.003	0.003	0.002	0.002	0.002
LDA	94.9	94.4	94.2	94.5	94.3	94.6	94.2	94.6	21.6	21.8	22.0	21.7	22.0	22.3	22.2	22.2
S-PT LDA	87.1	86.3	86.3	85.8	85.9	85.1	85.4	85.2	17.6	18.2	17.7	17.7	17.5	17.5	17.4	17.1
G-PT LDA																
kd = 0.0001	89.0	85.1	88.0	84.4	85.8	85.9	85.4	85.1	16.7	17.2	18.3	17.6	17.7	17.6	17.6	17.1
kd = 0.2	84.0	84.7	83.8	83.2	84.6	84.4	82.1	81.6	16.2	15.7	16.4	16.3	15.9	16.1	15.4	15.1
kd = 0.4	80.6	79.6	81.7	79.3	75.5	73.2	71.7	70.8	14.3	14.0	13.8	12.4	11.9	10.9	10.8	10.2
kd = 0.6	69.6	74.5	69.5	64.2	59.4	52.9	52.8	47.7	9.9	10.1	8.4	8.4	6.8	6.0	5.6	5.2
kd = 0.8	50.7	49.4	30.7	27.4	21.2	19.8	16.7	12.7	4.8	4.0	3.0	2.7	2.4	2.2	1.8	1.6
kd = 1.0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

With kd = 1, G-PT PCA has an EER near zero (0.005%), and G-PT LDA can reduce EER to zero. This means no one can pass the face authentication system without a correct password under G-PT LDA with kd $\geqslant$ 1. Table 3 shows FRR of zero FAR and EER in detail.

- *Scenario* 3
  An attacker who obtained users' passwords attempts to receive authentication similar to other users using his/her face and other users' passwords. The attacker is referred to as a "face imposter."
- *Experiment* 3
  For 194 classes, we performed experiments on G-PT PCA and G-PT LDA with different kd values. Instead of correct faces, we input different users' faces, which were randomly selected. We repeated each experiment 100 times.
- *Result* 3
  Fig. 9 shows the distance distribution and the FAR of face imposters. Even with a different kd, the distance distribution and FAR of a face imposter are not different. When correct passwords were input, the face authentication system computed $\mathbf{t} = \mathbf{Q}_i \cdot \mathbf{P}_i\mathbf{x} = \mathbf{S}_i\mathbf{U}\mathbf{x}$. Since there is no influence by $\mathbf{S}_i$ upon the measured distance, the distance distribution and FARs of face imposters will be similar to those of imposters in the original PCA and LDA. In addition, there is no influence by kd. Table 4 shows FRR of zero FAR and EER in detail.

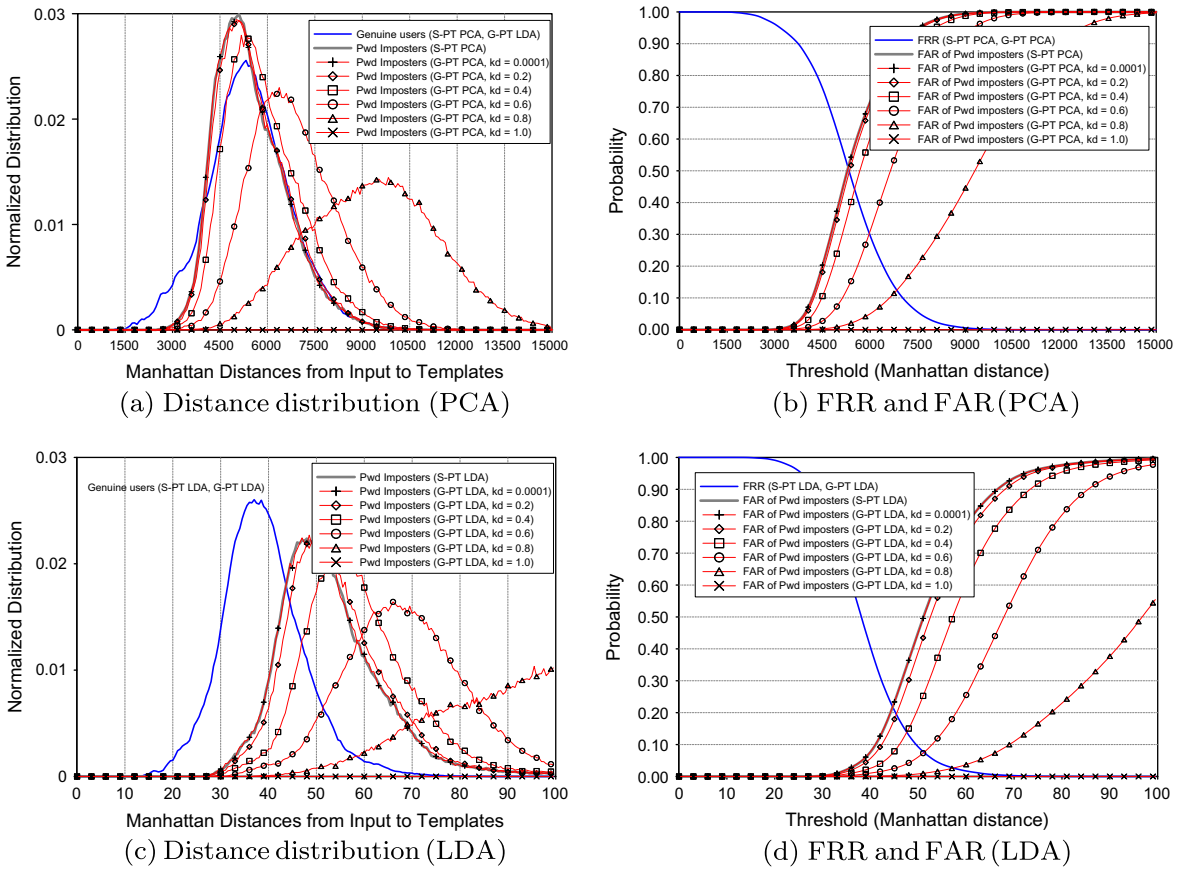


Fig. 8. Comparison between genuine users and password imposters (number of classes = 194).

Table 3
FRR of Zero FAR and the EER of Scenario 2 (password imposters) (%).

	FRR	EER
PCA (Scenario 1)	97.3	29.6
S-PT PCA	97.3	51.0
G-PT PCA		
kd = 0.0001	97.4	51.0
kd = 0.2	97.0	49.8
kd = 0.4	96.6	44.1
kd = 0.6	94.5	30.3
kd = 0.8	88.9	12.4
kd = 1.0	0.7	0.005
LDA (Scenario 1)	94.6	22.2
S-PT LDA	91.4	21.7
G-PT LDA		
kd = 0.0001	91.4	21.7
kd = 0.2	90.4	19.3
kd = 0.4	86.7	13.8
kd = 0.6	77.6	7.5
kd = 0.8	42.7	2.7
kd = 1.0	0	0

(■ = higher than base feature extraction scheme).

Fig. 10 shows ROC curves that summarize characteristics of our scheme. When coupled with PCA, our scheme outperforms PCA only when kd $\geqslant$ 0.6. With LDA, our scheme always shows better (or, at least, similar) performance compared to LDA.

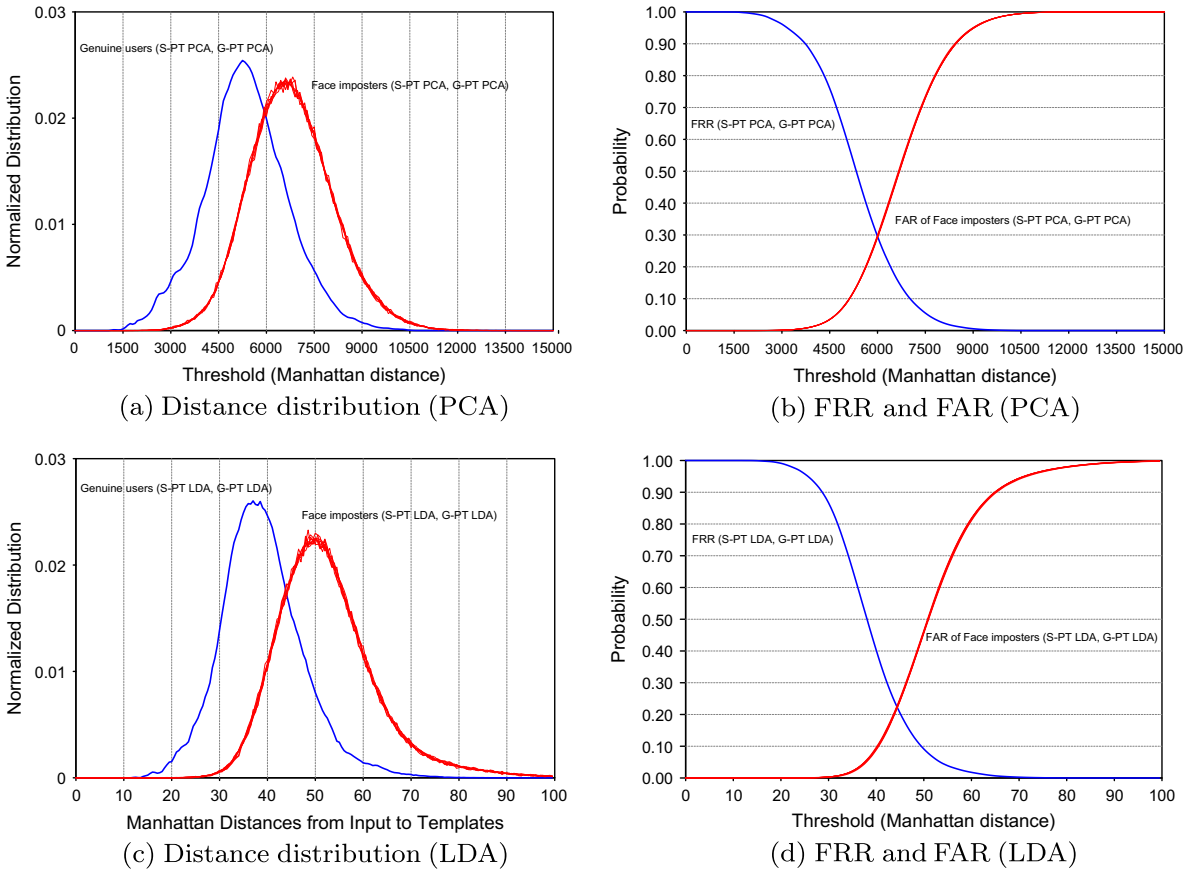


Fig. 9. Comparison between genuine users and face imposters (number of classes = 194).

Table 4
FRR of Zero FAR and the EER of Scenario 3 (face imposters) (%).

	FRR	EER
PCA (Scenario 1)	97.3	29.6
S-PT PCA	97.3	29.5
G-PT PCA		
kd = 0.0001	97.7	29.7
kd = 0.2	97.7	29.7
kd = 0.4	97.7	29.5
kd = 0.6	97.3	29.5
kd = 0.8	97.4	29.2
kd = 1.0	97.7	29.7
LDA (Scenario 1)	94.6	22.2
S-PT LDA	93.9	22.2
G-PT LDA		
kd = 0.0001	95.8	22.3
kd = 0.2	95.8	22.3
kd = 0.4	96.4	22.2
kd = 0.6	95.2	22.3
kd = 0.8	95.7	22.6
kd = 1.0	95.8	22.3

5.2. Face recovery

- *Scenario* 4
  An attacker who succeeds in invading the face authentication system wishes to know who has registered in the system.

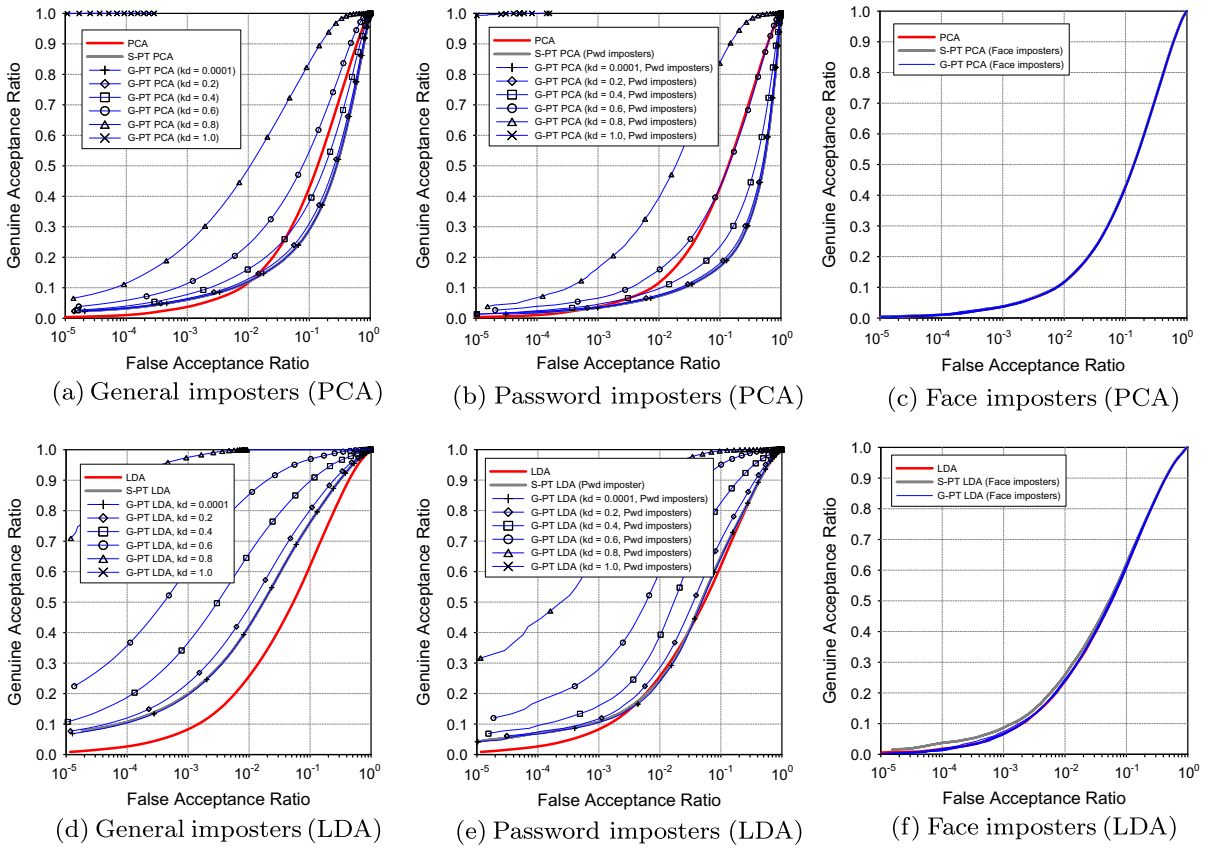


Fig. 10. Receiver operating characteristic curves (number of classes = 194).

- *Experiment* 4
  Utilizing Eqs. (3) and (10), we recovered the original face images from the stored templates in PCA/LDA, S-PT PCA/LDA, and G-PT PCA/LDA. To construct the projection matrices, we used 194 classes, where each class has 10 training images.
- *Result* 4
  As shown in Fig. 11, if an attacker obtains the global projection matrix $\mathbf{U}$ and templates $\mathbf{y}_i$ from an unprotected system, (s)he can recover the face images $\mathbf{x}'_i$. However, the attacker cannot extract any face images from the templates $\langle f_{r,\pi_i}(\mathbf{U}), \mathbf{Y}_i \rangle$ (S-PT) or $\langle \kappa_i, \mathcal{W}_i, \mathbf{Q}_i \rangle$ (G-PT) stored in a system that is protected by our scheme.

In S-PT PCA/LDA, the histogram remains similar in the recovered images despite the fact the recovered face image is scrambled. Even in this case, the histogram does not have sufficient information to recognize these images in a way people could recognize them. However, if the attacker assumes that these images are human faces, (s)he can attempt an exploiting attack against them. On the other hand, G-PT PCA/LDA erases the histogram information from the user templates, and thus the attacker cannot succeed in an exploiting attack in G-PT PCA/LDA.

5.3. Face and password dependencies

From the experiments, we observed that a larger kd increased the performance of G-PT PCA and G-PT LDA in the face authentication system. Paradoxically, this means that user passwords strongly affect the performance of G-PT LDA with larger kd.

Since both the S-PT and G-PT methods use two factors during the authentication procedure, false acceptances can occur in three cases: when only the password is wrong, when only the face is wrong, and when both the password and the face are wrong. FARs in the first and second cases indicate face and password dependencies, respectively.

Fig. 12 shows the ratios for 'FAR of password imposters' to 'FAR of face imposters' for different thresholds. If the ratio is close to 1, then both faces and passwords affect the performance to a similar extent. Please, note that the ratio does not mean how much effect passwords or faces have. As shown in Fig. 12, when the threshold is increased, the ratio becomes close to 1 in both PCA and LDA. In PCA, our scheme with small kd's ($\leqslant$0.4) is more dependent on faces. In LDA, by contrast, out scheme is always more dependent on passwords.

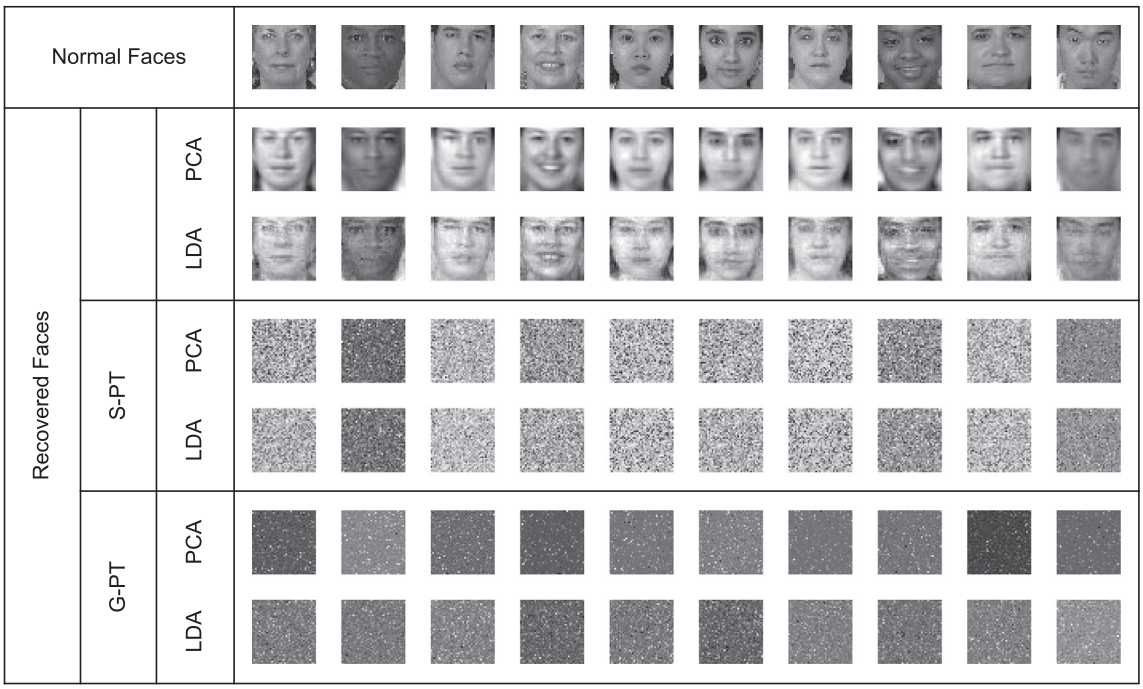


Fig. 11. Normalized recovered faces from the stored templates (gray FERET 1013–1022th classes, kd = 0.8).

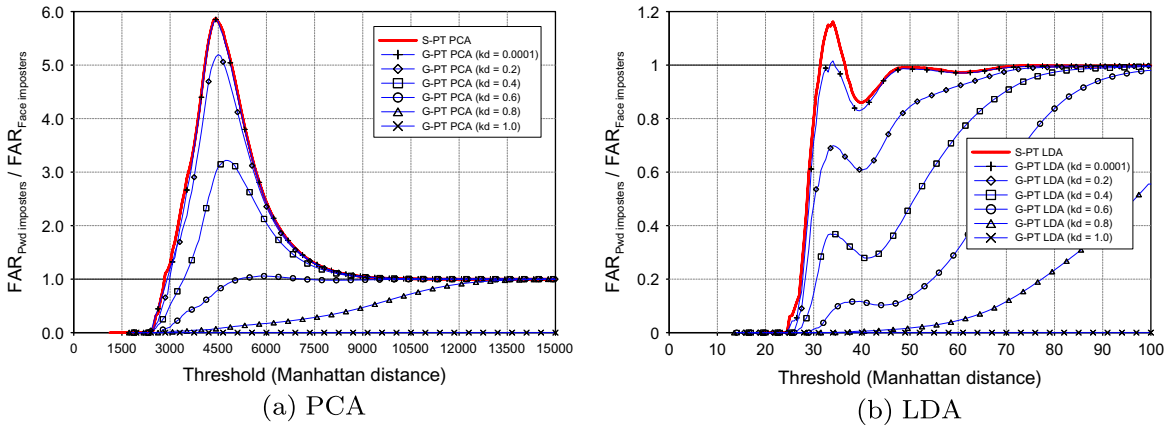


Fig. 12. $FAR_{Pwdimposters}$/ $FAR_{Faceimposters}$ ratio.

Since the threshold is often determined near EER or FRR of zero FAR, we must select an appropriate kd for the face authentication system. Therefore, to increase the threshold for a larger kd, better classifiers than LDA are required.

6. Security analysis

6.1. Dictionary attack

In a system that adopts our scheme, an attacker who has stolen a stored template can attempt to recover the corresponding user's face by finding the correct user password through a dictionary attack. The attacker can obtain the permuted face image $\mathbf{Px}'$ from the stolen template $\mathcal{T}_i = \langle \kappa_i, \mathcal{W}_i, \mathbf{Q}_i \rangle$ such that $\mathbf{Px}' = \mathbf{Q}_i^{+}\mathbf{S}_i\mathbf{y}_{(i,j)}$.

Moreover, from a guessed password π' (and the known salt κ_i), the attacker can compute $\mathbf{P}'$ and its inverse matrix. If $\mathbf{x}'' = (\mathbf{P}')^{-1}\mathbf{Px}'$ represents a face, the guessed password has a high probability of being correct. In order to confirm if $\mathbf{x}''$ is

a face, the attacker requires a face detection algorithm. Note that in the dictionary attack, salts prevent the attacker from preparing all possible permutation matrices.

Nevertheless, unlike the scheme in Kang and colleagues [20], this does not mean that system security is completely dependent on the minimum entropy of user passwords. Because of the isolation matrices $\{\mathbf{S}_i\}_{i\in[1\ldots c]}$, the attacker cannot obtain the projection matrix $\mathbf{U}$ and the feature vectors $\{\mathbf{Y}_i\}_{i\in[1\ldots c]}$ from the stolen templates. To remove $\mathbf{S}_i \in \mathbb{R}^{m\times m}$ from $\mathbf{S}_i\mathbf{U}$ and $\mathbf{S}_i\mathbf{Y}_i$, the attacker needs to search $m!/2$ cases on average. The only way to recover user face images is to try many independent dictionary attacks, as above. Until the attacker performs an actual dictionary attack, (s)he will not know if the user password is easy to break.

Our scheme enhances security against a dictionary attack by using salts and isolation matrices.

6.2. Exploiting attack

The scheme in Kang and colleages [20] is weak against an exploiting attack that determines who is registered in the face authentication system. However, the permutation matrices $\{\mathbf{P}_i\}_{i\in[1\ldots c]}$ in the G-PT method prevent an exploiting attack with the cost of affecting the performance of the face authentication. Since the permutation matrices render the distances of the feature vectors unpredictable, the G-PT method already holds an anti-exploiting feature, referred to as "generalization."

Another anti-exploiting method known as "normalization" can be employed. Normalization is employed to make the distances of all the face images identical. Despite its simplicity, normalization negatively affects the results of face authentication. Note that differences between all the face images (i.e., skin color) are used mainly to recognize face images. In other words, accuracy decreases, and a particular group of people will suffer from lower accuracy instead of a lower security threat. However, with development of a suitable method to recognize a face using the shape without color, normalization could be considered for adoption in a secure face authentication system.

7. Other issues

7.1. Adaptation to other face authentication systems

Our scheme works with linear projection systems – it uses a common linear projection matrix and feature vectors. Given feature vectors and a projection matrix, an attacker can successfully recover face images, as shown in Eq. (3). Thus, if a face authentication system does not hold feature vectors, as with support vector machines (SVMs) and neural networks, the system can avoid face recovery attacks. Naturally, reconstruction of face images in these systems is more difficult than in linear projection systems. The attacker, however, might still obtain sufficient information from the stored data to impersonate users. For similar reasons, many researchers have studied cancelable fingerprints, even though it is difficult to recover an owner's fingerprints from the minutia [49,50,55,58]. Therefore, even with unidirectionality of classifiers, face authentication systems require some method for protecting the information used in classification.

Some face authentication systems that do not employ feature vectors could be incorporated into our technique. For example, our scheme can work with SVMs [57]. SVMs are a set of supervised learning methods used for $\{-1,+1\}$ classifiers. In SVMs, due to the lack of feature vectors, it seems too difficult to recover training data. An SVM finds a hyperplane $\mathbf{w}\cdot\mathbf{x}-b=0$ that classifies the training data and defines a decision function $f(\mathbf{x})=\sum \alpha_i y_i\langle\mathbf{x}_i,\mathbf{x}\rangle+b$. By transforming the vector $\mathbf{w}$'s, and inputting a permuted face $\mathbf{Px}$ instead of a face $\mathbf{x}$, we can establish a secure face authentication system based on SVMs. For non-linear SVMs, however, the structure of the kernel $\langle\mathbf{x}_i,\mathbf{x}\rangle$ should be carefully considered. Non-linear SVMs use a "kernel trick," which is a justification for mapping a data set to a transformed feature space. Further studies should be performed on various systems, such as neural networks or hidden Markove models, to which our scheme cannot be directly applied.

On the other hand, hybrid schemes that combine our scheme and one or more existing techniques, such as biohashing, are also possible. In the matching routine (see Fig. 2), our scheme generates transformed feature vectors as intermediate values. By applying biohashing to the feature vectors, our scheme can simply be coupled with those techniques. However, for the hybrid schemes, you must carefully consider the influence of passwords (or user tokens).

7.2. Feature extraction and personal re-enrollment

Personal re-enrollment is a desirable property for face authentication systems. Human faces can change for various reasons, such as age, accidents, and surgical operations. In these cases, a user has to enroll his or her face again without changing other users' templates in the face authentication system. If our scheme uses the conventional classifiers, such as LDA, as the base feature extraction scheme, then it is difficult to support personal re-enrollment because the classifiers require sample data from all classes as the training data. Therefore, to support personal re-enrollment in our scheme, we need feature extraction that does not require sample data from all classes and reduces dimensions of input though the linear projection.

Random projection (RP) [6] is suited to these requirements. RP is already used for the schemes by Teoh and colleagues [52] and Feng and colleagues [9]. According to Johnson–Lindenstrauss's lemma [18], if points are projected onto a random

subspace of high dimension, the distances between the points are preserved. Goel and colleagues [12] investigated face recognition based on RP. In their paper, RP is slightly better than PCA for high dimensions (>80).

We leave this improved system based on RP for future research.

7.3. Comparison between existing schemes

As shown in Table 5, it is clear that user-supplied information, such as a password and a USB token, compensates for information loss from feature extraction. The existing schemes in Table 5 require user-supplied information or have the kind of helper data that helps authentication, and thus increases accuracy. Among helper data-based schemes, Kevenaar and colleagues' scheme [21], Kim and Toh's scheme [23], Feng and colleagues' scheme [9], and Li and colleagues' scheme [31] fail to archive zero EER. Basically, in Sutch and colleagues's scheme [51], the authors assume that the database stores the sketch that helps template reconstruction and key extraction, but they mention multi-factor authentication because of worries about low entropy of an extracted key and compromised sketch (i.e., helper data).

The biohashing-based schemes of zero EER (i.e., Ngo and colleagues [39] and Teoh and colleagues [53]) are coupled with the wavelet or Fourier transform. Except for them, only Lee and colleagues' scheme [28] and our scheme, which have permutation transformation as the core operation, achieve zero EER.

According to Kong and colleagues [25], the original biohashing scheme [13,39] delivered worse performance than the base feature extraction scheme without users' valid tokens. Teoh and colleagues' scheme [53] (improved biohashing) has the same problem as shown in Table 5. This implies that user tokens in the original and improved biohashing schemes have an overly strong impact on system performance, and thus an attacker who has valid tokens obtains more opportunities to pass authentication. Our scheme guarantees performance similar to the base feature extraction scheme, even when imposters had valid passwords.

7.4. Security and convenience: user supplied information and helper data

It is clear that user-supplied information inconveniences users. Face authentication systems increase convenience by storing user-supplied information in a database, but this can create a security problem from intruders. In the biohashing schemes, for example, an intruder who obtains a user token with a biohash string (user template) can successfully reconstruct features, according to Jain and collagues [16] and Kummel and Vielhauer [27]. Therefore, user-supplied information provides a simple way to guarantee a sufficient level of security by demanding that users themselves secure the user-supplied information.

The helper data-based schemes, such as those in Kevennar and colleagues [21] and Lu and colleagues [31], seem to be free from this problem, but they suffer from other possible problems. The first is low entropy of biometric keys. For example, the

Table 5
Summarized experimental results of existing schemes for secure face authentication.

Author(s)	Applied schemes	Genuine		Stolen face		Stolen key		Dataset (classes/poses)	User input
		FRR	EER	FRR	EER	FRR	EER		
Goh and Ngo [13]	PCA, BH	1.6	0.37	–	–	–	–	Face94 (153/20)	✓
Ngo et al. [39]	LDA, BH	0.06	0.03	–	–	–	–	FERET (200/5)	✓
	WT, BH	0	0	–	–	–	–	FERET (200/5)	✓
Kevenaar et al. [21]	Gabor kernel, ECC	35	1.5	–	–	N/A	N/A	FERET (237/4.9)	
		3.5	0.3	–	–	N/A	N/A	Caltech (24/14.4)	
Kang et al. [20]	PCA, S-PT	95.1	39.3	97.3	51.0	97.3	29.5	FERET (194/11)	✓
	LDA, S-PT	85.2	17.1	91.4	21.7	93.9	22.2	FERET (194/11)	✓
Lumini and Nanni [33]	DCT, improved BH	0	0	–	–	<40	<4	ORL (40/10)	✓
Teoh et al. [52]	LDA, RMQ (BH)	0.4	≈0	>2	–	>25	–	FERET (400/6)	✓
Teoh et al. [53]	FT, improved BH	0	0	0	0	≈90	15.9	FERET (400/6)	✓
	PCA, improved BH	–	2.2	–	2.1	>90	26.8	FERET (400/6)	✓
Sutcu et al. [51]	SVD, Sketch	≈0	≈0	–	–	–	–	Face94 (152/20)	(✓)
Kim and Toh [23]	mRP, Weighting	–	18.5	–	–	N/A	N/A	AR (102/26)	
Lee et al. [28]	LDA/PCA, PT	0	0	–	–	–	–	Self (55/20)	✓
Feng et al. [9]	RP, DP transform	–	6.8	–	–	N/A	N/A	CMU PIE (68/105)	
		–	8.6	–	–	N/A	N/A	FERET (250/4)	
Lu et al. [31]	PCA, BCH (ECC)	79.4	–	–	–	N/A	N/A	CMU PIE (68/21)	
Saini and Sinha [46]	FT, JPS, BH	9	4.5	–	–	–	–	Self, ORL, IFD	✓
Our scheme	PCA, G-PT	23.9	0.002	0.7	0.005	97.3	29.2	FERET (194/11)	✓
	LDA, G-PT	0	0	0	0	95.2	22.2	FERET (194/11)	✓

(FRR = 0-FAR, ■ = higher than base feature extraction scheme)

biometrics key in Lu and colleagues' scheme [31] are about 16–36 bits. In practice, brute force attack works effectively at this complexity. However, because of the uncertainty in face images, it is difficult to make biometric keys sufficiently long. The second problem is that helper data may reveal too much information. Ignatenko and Willems [15] investigate information leakage in fuzzy commitment-based schemes. They mention that Kevenaar and colleagues' scheme [21] does not satisfy the criteria for secure conditions of fuzzy commitment. Al-Assam and Jassim [2] show biometric keys are not sufficiently secure for face applications. In that sense, Sutch and colleagues' consideration of multi-factor authentication is reasonable [51].

In a conservative view of security, helper data has less value than user-supplied information. Indeed, helper data has to be carefully designed [16].

7.5. Applications for two-factor face authentication

Basically, our face authentication scheme is applicable in all areas that require high-level access control. In those areas, single-factor authentication is more unusual. Fingerprints, voiceprints, or the iris can be used at the same time as ID cards and passwords. In that sense, our scheme can easily be applied to automatic teller machines equipped with a security camera. Face authentication requires unmasked faces (e.g., no sunglasses), and thus automatic teller machines can prevent illegal withdrawals by thieves or robbers if our scheme is applied.

Unlike personal computers, most smartphones have small and cheap cameras as basic equipment. In practice, the latest Android operating system supports smartphone unlocking using face authentication. Therefore, there is no obstruction to performing our face authentication scheme with smartphones. Today, many banks aggressively offer mobile banking services using smartphones. To transfer money using mobile banking, users input various types of secrets such as login password, transfer password, one-time password number, and secret card number. We expect that by applying our face authentication scheme to mobile banking, banks can enhance security.

8. Conclusion

For secure face authentication, we introduced a two-factor face authentication scheme. Unlike the scheme in Kang and colleagues [20], which involves specially designed permutation matrices generated from user passwords, our scheme can be employed with general invertible matrices as permutation matrices. With additional secure factors (i.e. secure matrices and general permutation matrices), our scheme guarantees a safer environment than that of Kang and colleagues [20]. Permutation transformation does not require rebuilding the entire database when canceling information. Furthermore, for a higher security level, the system can be enhanced with hardware tokens and dummy templates.

The usage of external secure information (i.e., user-supplied information) can be permitted due to the technical difficulties of secure face authentication. However, as confirmed in our experiment results, external information considerably affects the overall face authentication ratio. For face authentication with high security, the effectiveness of external information is sufficient. Therefore, our scheme is more suitable for secure face authentication systems than for face recognition systems. In order to achieve a secure face authentication system with better performance, research should be conducted on good face classifiers that can overcome the limitations related to face feature extraction and the number of sample classes.

It is clear that our scheme works well with any linear projection scheme; its performance is on par with PCA and LDA, and we expect that our scheme can be applied to many non-feature systems. In addition, a hybrid scheme coupled with biohashing and the re-enrollment-enabled scheme based on RP are worth consideration. However, according to the structures, our scheme may not be applicable, or other additional methods may be required. We leave these issues to future studies.

Acknowledgment

This work was supported by Inha University, Republic of Korea.

References

[1] A.F. Abate, M. Nappi, D. Riccio, G. Sabatino, 2d and 3d face recognition: a survey, Pattern Recognit. Lett. 28 (2007) 1885–1906.
[2] H. Al-Assam, S. Jassim, Security evaluation of biometric keys, Comput. Secur. 31 (2012) 151–163.
[3] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, Face recognition by independent component analysis, IEEE Trans. Neural Netw. 13 (2002) 1450–1464.
[4] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Eigenfaces vs. fisherfaces: recognition using class specific linear projection, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1997) 711–720.
[5] B. Biggio, Z. Akhtar, G. Fumera, G. Marcialis, F. Roli, Security evaluation of biometric authentication systems under real spoofing attacks, Biometrics IET 1 (2012) 11–24.
[6] E. Bingham, H. Mannila, Random projection in dimensionality reduction: applications to image and text data, in: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining KDD '01, ACM, New York, NY, USA, 2001, pp. 245–250.
[7] M. Braithwaite, U.C. von Seeln, J. Cambier, J. Daugman, R. Glass, R. Moore, I. Scott, Application-specific biometric templates, in: IEEE Workshop on Automatic Identification Advanced Technologies, 2002, pp. 167–171.
[8] L.-F. Chen, H.-Y.M. Liao, M.-T. Ko, J.-C. Lin, G.-J. Yu, A new lda-based face recognition system which can solve the small sample size problem, Pattern Recognit. 33 (2000) 1713–1726.
[9] Y.C. Feng, P.C. Yuen, A.K. Jain, A hybrid approach for face template protection, SPIE Defense and Security Symposium, vol. 102, SPIE, 2008, pp. 169–177.

[10] Y.C. Feng, P.C. Yuen, A.K. Jain, A hybrid approach for generating secure and discriminating face template, IEEE Trans. Inform. Forensics Secur. 5 (2010) 103–117.
[11] S.L. France, J.D. Carroll, H. Xiong, Distance metrics for high dimensional nearest neighborhood recovery: compression and normalization, Inform. Sci. 184 (2012) 92–110.
[12] N. Goel, G. Bebis, A. Nefian, Face recognition experiments with random projection, in: Proc. SPIE 5779, Biometric Technology for Human Identification II, 2005, pp. 426–437.
[13] A. Goh, D.C. Ngo, Computation of cryptographic keys from face biometrics, in: A. Lioy, D. Mazzocchi (Eds.), Communications and Multimedia Security, Lecture Notes in Computer Science, vol. 2828, Springer, 2003, pp. 1–13.
[14] Y. Huang, J. Zhao, Y. Liu, S. Luo, Q. Zou, M. Tian, Nonlinear dimensionality reduction using a temporal coherence principle, Inform. Sci. 181 (2011) 3284–3307.
[15] T. Ignatenko, F.M.J. Willems, Information leakage in fuzzy commitment schemes, IEEE Trans. Inform. Forensics Secur. 5 (2010) 337–348.
[16] A.K. Jain, K. Nandakumar, A. Nagar, Biometric template security, EURASIP J. Adv. Signal Process. 2008 (2008) 1–17.
[17] Z. Jin, J.-Y. Yang, Z.-S. Hu, Z. Lou, Face recognition based on the uncorrelated discriminant transformation, Pattern Recognit. 34 (2001) 1405–1416.
[18] W.B. Johnson, J. Lindenstrauss, Extensions of Lipschitz mapping into Hilbert space, in: Conf. in Modern Analysis and Probability, Contemporary Mathematics, vol. 26, American Mathematical Society, 1984, pp. 189–206.
[19] A. Juels, M. Wattenberg, A fuzzy commitment scheme, in: ACM Conference on Computer and Communications Security, 1999, pp. 28–36.
[20] J. Kang, D. Nyang, K. Lee, Two factor face authentication scheme with cancelable feature, in: S.Z. Li, Z. Sun, T. Tan, S. Pankanti, G. Chollet, D. Zhang (Eds.), IWBRS, Lecture Notes in Computer Science, vol. 3781, Springer, 2005, pp. 67–76.
[21] T.A.M. Kevenaar, G.-J. Schrijen, M. van der Veen, A.H.M. Akkermans, F. Zuo, Face recognition with renewable and privacy preserving binary templates, in: Fourth IEEE Workshop on Automatic Identification Advanced Technologies, October 2005, pp. 21–26.
[22] Y. Kim, A.B.J. Teoh, K.-A. Toh, A performance driven methodology for cancelable face templates generation, Pattern Recognit. 43 (2010) 2544–2559.
[23] Y. Kim, K.-A. Toh, A method to enhance face biometric security, in: First IEEE International Conference on Biometrics: Theory, Applications, and Systems (BTAS 2007), 2007a, pp. 1–6.
[24] Y. Kim, K.-A. Toh, Sparse random projection for efficient cancelable feature extraction, in: IEEE Conference on Industrial Electronics and Applications, 2007b, pp. 2139–2144.
[25] A. Kong, K.-H. Cheung, D. Zhang, M. Kamel, J. You, An analysis of biohashing and its variants, Pattern Recognit. 39 (2006) 1359–1368.
[26] B.V.K.V. Kumar, M. Savvides, C. Xie, Correlation pattern recognition for face recognition, Proc. IEEE, vol. 94, IEEE, 2006, pp. 1963–1976.
[27] K. Kümmel, C. Vielhauer, Reverse-engineer methods on a biometric hash algorithm for dynamic handwriting, in: Proceedings of the 12th ACM Workshop on Multimedia and Security MMSec'10, ACM, New York, NY, USA, 2010, pp. 67–72.
[28] Y. Lee, Y. Lee, Y. Chung, K. Moon, One-time templates for face authentication, in: ICCIT '07: Proceedings of the 2007 International Conference on Convergence Information Technology, IEEE Computer Society, Washington, DC, USA, 2007, pp. 1818–1823.
[29] R. Lotlikar, R. Kothari, Fractional-step dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 22 (2000) 623–627.
[30] G.-F. Lu, Y. Wang, Feature extraction using a fast null space based linear discriminant analysis algorithm, Inform. Sci. 193 (2012) 72–80.
[31] H. Lu, K. Martin, F. Bui, K. Plataniotis, D. Hatzinakos, Face recognition with biometric encryption for privacy-enhancing self-exclusion, in: 2009 16th International Conference on Digital Signal Processing, July, pp. 1–8.
[32] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, Regularized discriminant analysis for the small sample size problem in face recognition, Pattern Recognit. Lett. 24 (2003) 3079–3087.
[33] A. Lumini, L. Nanni, An improved biohashing for human authentication, Pattern Recognit. 40 (2007) 1057–1065.
[34] A.M. Martínez, A.C. Kak, Pca versus lda, IEEE Trans. Pattern Anal. Mach. Intell. 23 (2001) 228–233.
[35] T. Matsumoto, H. Matsumoto, K. Yamada, S. Hoshino, Impact of artificial gummy fingers on fingerprint systems, Proceedings of SPIE, vol. 4677, SPIE, 2002, pp. 275–289.
[36] E.H. Moore, On the reciprocal of the general algebraic matrix, Bull. Am. Math. Soc. 26 (1920) 394–395.
[37] L. Nanni, A. Lumini, Empirical tests on biohashing, Neurocomputing 69 (2006) 2390–2395.
[38] L. Nanni, A. Lumini, Local binary patterns for a hybrid fingerprint matcher, Pattern Recognit. 41 (2008) 3461–3466.
[39] D.C.L. Ngo, A.T.B. Jin, A. Goh, Biometric hash: high-confidence face recognition, IEEE Trans. Circ. Syst. Video Technol. 16 (2006) 771–775.
[40] T. Ojala, M. Pietikainen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recognit. 29 (1996) 51–59.
[41] R. Penrose, A generalized inverse for matrices, Proc. Cambridge Philos. Soc. 51 (1955) 406–413.
[42] V. Perlibakas, Distance measures for pca-based face recognition, Pattern Recognit. Lett. 25 (2004) 711–724.
[43] P.J. Phillips, H. Moon, P. Rauss, S.A. Rizvi, The feret evaluation methodology for face-recognition algorithms, in: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97), IEEE Computer Society, Washington, DC, USA, 1997, p. 137.
[44] P.J. Phillips, H. Wechsler, J. Huang, P.J. Rauss, The feret database and evaluation procedure for face-recognition algorithms, Image Vision Comput. 16 (1998) 295–306.
[45] N.K. Ratha, J.H. Connell, R.M. Bolle, Enhancing security and privacy in biometrics-based authentication systems, IBM Syst. J. 40 (2001) 614–634.
[46] N. Saini, A. Sinha, Optics based biohashing using joint transform correlator, Opt. Commun. 283 (2010) 894–902.
[47] M. Savvides, B.V.K.V. Kumar, P.K. Khosla, Cancelable biometric filters for face recognition, in: 17th International Conference on the Pattern Recognition (ICPR'04), vol. 3, IEEE Computer Society, Washington, DC, USA, 2004, pp. 922–925.
[48] A. da Silva Pinto, H. Pedrini, W. Schwartz, A. Rocha, Video-based face spoofing detection through visual rhythm analysis, in: 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), 2012, pp. 221–228.
[49] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, B.V. Kumar, Biometric encryption using image processing, in: R.L. van Renesse (Ed.), Optical Security and Counterfeit Deterrence Techniques II, Proceedings of SPIE, vol. 3314, SPIE, 1998, pp. 178–188.
[50] C. Soutar, G.J. Tomko, Secure private key generation using a fingerprint, in: CardTech/SecurTech Conference, Applications in Action, vol. 1, CardTech/SecurTech, Inc, 1996, pp. 245-252.
[51] Y. Sutcu, Q. Li, N. Memon, Protecting biometric templates with sketch: Theory and practice, IEEE Trans. Inform. Forensics Secur. 2 (2007) 503–512.
[52] A.B. Teoh, A. Goh, D.C. Ngo, Random multispace quantization as an analytic mechanism for biohashing of biometric and random identity inputs, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 1892–1901.
[53] A.B. Teoh, Y.W. Kuan, S. Lee, Cancellable biometrics and annotations on biohash, Pattern Recognit. 41 (2008) 2034–2044.
[54] M.A. Truk, A.P. Pentland, Face recognition using eigenfaces, J. Cogn. Neurosci. 3 (1991) 71–86.
[55] U. Uludag, S. Pankanti, A.K. Jain, Fuzzy vault for fingerprints, in: T. Kanade, A.K. Jain, N.K. Ratha (Eds.), AVBPA, vol. 3546, Springer, 2005, pp. 310–319.
[56] U. Uludag, S. Pankanti, S. Prabhakar, A.K. Jain, Biometric cryptosystems: issues and challenges, Proc. IEEE 6 (2004) 948–960.
[57] V. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995.
[58] S. Yang, I.M. Verbauwhede, Secure fuzzy vault based fingerprint verification system, in: The Thirty-Eighth Asilomar Conference on Signals, Systems, and Computers, vol. 1, IEEE Signal Processing Society, 2004, pp. 577-581.
[59] H. Yin, W. Huang, Adaptive nonlinear manifolds and their applications to pattern recognition, Inform. Sci. 180 (2010) 2649–2662.
[60] H. Yu, J. Yang, A direct lda algorithm for high-dimensional data – with application to face recognition, Pattern Recognit. 34 (2001) 2067–2070.