A Cloud-assisted Passenger Authentication Scheme for Japan Rail Pass Based on Image Morphing

Yanjun Liu and Chin-Chen Chang
(Corresponding author: Chin-Chen Chang)

Department of Information Engineering and Computer Science, Feng Chia University
No. 100, Wenhua Road, Xitun District, Taichung 40724, Taiwan
(Email: alan3c@gmail.com)

(Received July 27, 2017; revised and accepted Oct. 21, 2017)

Abstract

The Japan rail (JR) pass is the most economical means to travel around Japan through public transportations. Therefore, the passenger authentication for JR pass has become a very typical and popular smart-card based application. However, a traditional passenger authentication scheme for the JR pass has two major problems: 1) The passenger must present his/her passport to the attendant each time for the authentication, increasing the probability of losing the passport; 2) The passenger's passport number, which is printed on the JR pass for authentication, may reveal the passenger's personal information. To overcome these security and privacy weaknesses, we innovatively propose a cloud-assisted passenger authentication scheme for the JR pass based on image morphing technique. Analyses show that the method of our proposed scheme is quite simple to implement and can resist well-known attacks.

Keywords: Cloud-assisted; Image Morphing; Japan Rail (JR) Pass; Passenger Authentication; Security

1 Introduction

Nowadays, user authentication is used extensively in many applications in the field of information security. The remote user authentication mechanism was originally proposed by Lamport [8] in 1981. There are two parties in the authentication mechanism, *i.e.*, a user and a server. The user is allowed to set a password and transmits it to the server secretly for registration. When the user wants to gain some services from the server, he/she must be requested to provide the password to the server to authenticate his/her identity. Unfortunately, the server must maintain a verifier table [8] that preserves some personal information of the users. When the number of users increases drastically, it requires huge storage space for the verifier table. Moreover, the use of the verifier table risks a severe security problem such that the theft of the verifier table can induce a leak of users' private information. Given that a smart card can enhance security as well as efficiency, many authentication schemes [7,9,10] based on smart card have been proposed.

In a smart card-based authentication scheme, each user is assigned a smart card rather than central storage for confidential information as used in conventional schemes. Each smart card stores some holders' personal information, thereby significantly diminishing the risk of information leak and increasing the storage efficiency [10]. The smart card can also preserve additional information used for tamper-proof to increase security.

The smart card-based authentication consists of a card authentication process and a user authentication process [9]. After the smart card is registered to the server, the registered information is stored in the smart card and then transmitted to the terminal to judge the legality of the card. It is considered more secure since the card authentication process can be achieved only between the smart card and the terminal rather than involving the server. However, even if the card authentication is passed, it cannot ensure that the card holder is the true user. Thus, the user should be authenticated after the card authentication. Biometrics information [6,11] such as facial features, fingerprint and iris of the user is usually requested to provide for the user authentication. Because biometrics information of two different people cannot be identical, the illegal user can be immediately identified if he/she cannot offer the biometrics information unique to the true user's body. For convenience, the biometrics information of the true user is often stored or printed on the smart card. When the card is used, a human operator performs a face-to-face authentication of the card holder by comparing the biometrics information offered by the card holder with that of the true user. If the biometrics information are the same, the human operator can be convinced that the card holder is legal.

The concept of smart card-based authentication can be used in many applications in our real life thanks to its advantages in high security and low cost. The authentication for Japan rail (JR) pass [1, 2] is a very typical and popular application among these applications. Therefore, inspired by the card user authentication, we will propose a novel method to authenticate users for the JR pass in this paper. The JR pass [1], provided by the Japan Railways Group, is the most economical means to travel around Japan through public transportations such as trains, buses and ferries. However, the JR pass can only be used by overseas tourists for sight-seeing and Japanese nationals living outside of Japan who meet some particular requirements. Only eligible passengers can purchase 7, 14 or 21 day length JR pass to plan their trip. In fact, a JR pass can be regarded as a smart card. Therefore, how to efficiently authenticate the eligible passenger when the JR pass is used becomes a crucial issue.

Now let us take a look at the way that a traditional passenger authentication scheme for the JR pass conducts [2]. An eligible passenger can purchase a JR pass at a sales office at a JR station or a travel agency by presenting his/her passport to the attendant. Then, the passenger is issued with a JR pass on which the passport number is printed. Before every boarding, the validity of the passenger should be authenticated by an attendant at a manned ticket gate at a JR station. For the authentication, the passenger is requested to present his/her JR pass and passport together to the attendant. The attendant then confirms that the passenger is the owner of the passport and the passport number on the JR pass is the same as that on the passport. The passenger is permitted to board only if the authentication is passed.

Although the above mentioned passenger authentication is simple to implement, it leads to two security problems. First, the passenger must present his/her passport to the attendant each time for the authentication, which means that the passenger needs to carry the passport all the time so that the passport is easy to get lost. Second, the passport number should be printed on the JR pass to perform the authentication. Unfortunately, the disclosure of the passenger's personal information (*i.e.*, passport number) may happen if the JR pass is stolen, lost or discarded when it expires. To overcome these weaknesses, we will innovatively propose a more secure passenger authentication scheme for the JR pass based on image morphing. Image morphing [3] is a technique which is originally used to produce special visual effects in movie industry. It creates a morphed image by using two images, one called source image and the other called target image. The created morphed image looks like both the source image and the target image if they have similar structures. This impressive feature makes image morphing a newly widespread approach in the area of authentication [12–16]. In this paper, a cloud-assisted passenger authentication scheme for the JR pass using image morphing is proposed to increase the security efficiently. To the best of our knowledge, the proposed scheme is the first authentication scheme for the JR pass that combines image morphing and cloud storage techniques. The unique characteristics of the proposed scheme are listed below:

1) To effectively protect the passenger's privacy, the passenger's passport number is no longer printed on the JR pass; instead, facial features of the passenger are employed for authentication. In particular, a morphed image is generated by the face image of the passenger and a pre-selected reference image through morphing, thereby hiding the face image of the passenger into the morphed image.

2) The generated morphed image is stored on the cloud storage, which efficiently avoids the disclosure of the passenger's personal information.

3) The authentication for the JR passenger is quite simple. When the JR pass is used, only an image de-morphing process is performed to verify the validity of the passenger. An attendant at a manned ticket gate uses the morphed image stored on the cloud storage and the reference image to restore the face image of the passenger through de-morphing. This method can significantly increase both security and convenience since the passenger's passport never needs to be used during the authentication.

The rest of the paper is organized as follows. In Section 2, we offer an overview of image morphing and some user authentication schemes based on image morphing. Section 3 proposes a cloud-assisted passenger authentication scheme for the JR pass using image morphing. Section 4 analyzes the correctness and security of the proposed scheme. Ultimately, our conclusions are given in Section 5.

2 Preliminaries

In this section, we introduce some background knowledge before presenting the proposed scheme. We first give the basic knowledge of image morphing, and then describe some related authenticated scheme using image morphing.

2.1 Image Morphing and De-Morphing

Image morphing technique [3] uses a source image and a target image to create a morphed image that looks like both the source image and the target image if they have similar structures. Image de-morphing, as its name implies, is the reverse of image morphing that restores the source image (or the target image) from the morphed image and the target image (or the source image). In the following, we will briefly introduce how image morphing and de-morphing work, respectively.

The source image and the target image are denoted as I_s and I_t with the size of $N_1 \times N_2$, respectively. Now we present how to generate the morphed image I_{st}^m using I_s and I_t.

2.1.1 Image Morphing Process [3]

Step 1: Choose $n (n \in N)$ control pixel pairs. One pixel chosen from I_s and its corresponding pixel chosen from I_t constitute a control pixel pair. Assume that (x_i^s, y_i^s) and (x_i^t, y_i^t) for $i = 1, 2, \ldots, n$ denote the coordinates of i^{th} control pixel in I_s and I_t, respectively. The coordinates of all the selected control pixels in I_s and I_t, represented as matrices C_s and C_t, are shown as follows:

$$C_s = \begin{bmatrix} x_1^s & x_2^s & \ldots & x_n^s \\ y_1^s & y_2^s & \ldots & y_n^s \end{bmatrix} \qquad (1)$$

$$C_t = \begin{bmatrix} x_1^t & x_2^t & \ldots & x_n^t \\ y_1^t & y_2^t & \ldots & y_n^t \end{bmatrix} \qquad (2)$$

Step 2: Calculate horizontal and vertical distances of control pixel pairs in I_s and I_t. Horizontal-distance vector D_1 and vertical-distance vector D_2 are computed as follows:

$$D_1 = \begin{bmatrix} x_1^s - x_1^t & x_2^s - x_2^t & \cdots & x_n^s - x_n^t \end{bmatrix} \qquad (3)$$

$$D_2 = \begin{bmatrix} y_1^s - y_1^t & y_2^s - y_2^t & \cdots & y_n^s - y_n^t \end{bmatrix} \qquad (4)$$

Step 3: Calculate horizontal and vertical distances of all corresponding pixel pairs in I_s and I_t. In order to retrieve the distances of all corresponding pixel pairs from n control pixel pairs, linear interpolations are made on D_1 and D_2, and then interpolation matrices B_1 and B_2 with the size of $N_1 \times N_2$ are generated. Obviously, the component in matrix B_1, denoted as $b_1(x, y)$, represents the horizontal distance between the pixel (x, y) in I_s and its corresponding pixel in I_t. Accordingly, the component in matrix B_2, denoted as $b_2(x, y)$, represents the vertical distance between the pixel (x, y) in I_s and its corresponding pixel in I_t. For more detailed information about the implementation of interpolation, interested readers can refer to Ref. [14].

Step 4: Warp the source image and the target image. Assume that $\alpha (0 \leq \alpha \leq 1)$ is the morphing rate. To warp the source image I_s, the pixel (x, y) in I_s is shifted by $[\alpha b_1(x, y)]$ in the horizontal direction and by $[\alpha b_2(x, y)]$ in the vertical direction. The following equation accurately describes the warping process for the pixel (x, y) in I_s:

$$p_s^w(x, y) = p_s(x + [\alpha b_1(x, y)], y + [\alpha b_2(x, y)]), \qquad (5)$$

where $p_s(x, y)$ and $p_s^w(x, y)$ are the gray values of pixel (x, y) in I_s before and after warping, and $[h]$ is the rounding of h. Then, the warped source image I_s^w is generated when the shift of all the pixels in I_s completes.

The target image I_t is warped by a similar means. To warp the target image I_t, the pixel (x, y) in I_t is shifted by $[(1-\alpha)b_1(x, y)]$ in the horizontal direction and by $[(1-\alpha)b_2(x, y)]$ in the vertical direction. The warping process for the pixel (x, y) in I_t is defined as follows:

$$\begin{aligned} p_t^w(x, y) = p_t(x + [(1-\alpha)b_1(x, y)], \\ y + [(1-\alpha)b_2(x, y)]), \end{aligned} \qquad (6)$$

where $p_t(x, y)$ and $p_t^w(x, y)$ are the gray values of pixel (x, y) in I_t before and after warping. After that, the warped target image I_t^w is generated.

Step 5: Generate the morphed image. The morphed image I_{st}^m is created by using the warped source image I_s^w, the warped target image I_t^w and an appropriate morphing rate α as follows:

$$I_{st}^m = (1-\alpha)I_s^w + \alpha I_t^w. \qquad (7)$$

Image de-morphing is the reverse of image morphing that restores the source image (or the target image) from the morphed image and the target image (or the source image). In the following, we take the reconstruction of the source image as an example to explain the process of de-morphing. It is noticed that the de-morphing operation is on the assumption that we have already known the coordinates matrix C_s, the target image I_t, the coordinates matrix C_t, the morphed image I_{st}^m and the morphing rate α. First, horizontal-distance vector D_1 and vertical-distance vector D_2 for control pixel pairs from the source image I_s and the target image I_t are computed by Equations (3) and (4). Then, interpolation matrices B_1 and B_2 are obtained after linear interpolations are made on D_1 and D_2. After that, the warped target image I_t^w is obtained by Equation (6). According to Equation (7), the warped source image I_s^w can be computed as:

$$I_s^w = \frac{I_{st}^m - \alpha I_t^w}{1-\alpha} \qquad (8)$$

Finally, every pixel in I_s^w can easily return to the original location in I_s by the following operation:

$$p_s(x, y) = p_s^w(x - [\alpha b_1(x, y)], y - [\alpha b_2(x, y)]), \qquad (9)$$

and then the restored source image $I_{st \to s}^{de}$ is generated immediately.

An example of the processes of image morphing and de-morphing is shown in Figure 1. The source image and the target image, selected from Yale face database [5], are shown in Figures 1 (a) and 1 (b), respectively. The Morphed image created by Figures 1 (a) and 1 (b) with the morphing rate $\alpha = 0.5$ is illustrated in Figure 1 (c). Figure 1 (d) illustrates the restored source images by de-morphing.

2.2 Related Work

In recent years, image morphing is used extensively in user authentication schemes [12–16]. Zhao and Hsieh [16] proposed a card user authentication scheme using the user's

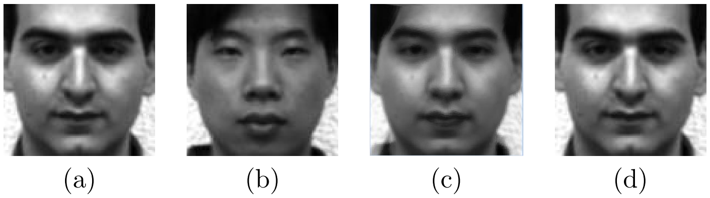


Figure 1: Image morphing and de-morphing: (a) source image; (b) target image; (c) morphed image ($\alpha = 0.5$); (d) recovered source image ($\alpha = 0.5$)

face image as important information for authentication. In their scheme, a morphed image is created by the face image of the card user and a reference image and then is printed on the card. Thus, the face image of the card user is actually concealed into the morphed image. When the card is used, a human operator should authenticate the identity of the card holder. The operator uses the morphed image and the reference image to recover a face image through de-morphing. If the recovered image is identical to the face image of the card user, it indicates that the card holder is legal and he/she can use the card to obtain required services. In addition, the authors extended the image morphing to the case when $l(l > 1)$ reference images are used and called it generalized image morphing. A user authentication scheme using generalized image morphing was also proposed and the security was analyzed thoroughly.

One of the most crucial issues in image morphing is to increase the visual effect of the morphed image [12, 15]. Thus, many algorithms focusing on the optimization of the selection of control pixel pairs from the source image and target image are proposed to achieve this goal. Also, these improved image morphing techniques are applied to the design of authentication schemes. In 2013, Mao *et al.* [12] proposed an edge directed automatic control pixel selection algorithm for better edge detection during morphing instead of manual selection. The experimental results showed that the new algorithm can increase both accuracy and efficiency of morphing. Zhao *et al.* [15] used an interactive genetic algorithm (IGA) to build an appropriate feature point set (FPS) to make more natural-looking morphed images. Thus, their previously proposed card user authentication scheme [16] can be enhanced by using the improved morphing method. Later, in 2015, Mao *et al.* [13, 14] innovatively proposed two authentication schemes based on image morphing. In [13], Mao *et al.* first presented a source-based image morphing (SBIM) algorithm that selects control pixels only in the source image. Accordingly, the de-morphing employs the coordinates of control pixels only in the source image to recover the original source image. Furthermore, a novel proxy user authentication (PUA) authentication scheme based on SBIM was proposed. The scheme can authenticate both a primary user and a proxy user who acts as a deputy of the primary user via image exchange. The key agreement scheme proposed in [14] is also implemented by image exchange. A communication user uses a pre-assigned secret image as the source image and another selected image as the target image to generate a morphed image, and then sends it to the receiver. The receiver reconstructs the target image from the morphed image and the source image by de-morphing. A secret session key can be established if the reconstructed target image is the same as the original target image. Moreover, the key agreement scheme can resist both active and passive attacks.

3 Proposed Scheme for JR Pass

In this section, we propose a novel passenger authentication scheme for the JR pass, which innovatively adopts image morphing and cloud storage techniques to significantly enhance the security compared to traditional JR pass authentication mechanisms. Our proposed scheme involves four entities, *i.e.*, a passenger, an attendant with a terminal at a sales office, an attendant with a terminal at a manned ticket gate and a set of cloud storage servers. An eligible passenger can purchase a JR pass with his/her passport from an attendant at a sales office. Only an assigned JR pass number appears on the pass for privacy protection and is used for authentication. Meanwhile, a morphed image that hides the face image of the passenger is generated and then stored on a cloud storage server. When the passenger wants to board a vehicle, an attendant at a manned ticket gate must authenticate the validity of the passenger. According to the number printed on the JR pass, the attendant retrieves the morphed image corresponding to the passenger from the cloud storage servers, and then recovers the face image of the passenger by de-morphing the morphed image. The passenger is permitted to board only if the authentication is passed.

Our proposed scheme mainly consists of the pass buying phase and the passenger authentication phase. The following subsections elaborate both phases of the proposed scheme.

3.1 Definition of Notations

Before the detailed description of both phases, the definition of the notations used in the proposed scheme is given in Table 1.

Table 1: Notations used in the proposed scheme

Notation	Definition
I_j	Original image j
I_j^w	Warped image j
I_{jk}^m	Morphed image using I_j as the source image and I_k as the target image
$I_{jk\to j}^{de}$	Restored source image j from I_{jk}^m via de-morphing
C_j	Coordinates of the control pixels of I_j
$\alpha_j (0 \leq \alpha_j \leq 1)$	Morphing rate corresponding to I_j
M	Morphing function
DM	De-morphing function
PG_j	Passenger j whose face image is I_j
PID_j	Number of the JR pass held by PG_j
IND_j	Index of PID_j in the set of all sorted JR pass numbers
ADS	Attendant at a sales office
TS	Terminal at a sales office
ADG	Attendant at a manned ticket gate
TG	Terminal at a manned ticket gate
CSS	Cloud storage servers
L_{jk}^m	Location on CSS where I_{jk}^m is stored

3.2 Pass Buying Phase

In this phase, an eligible passenger purchases a JR pass from an attendant at a sales office. Different from the traditional JR pass, the passenger's passport number is no longer printed on it so that the disclosure of private information can be avoided efficiently. Instead, only an assigned, unique number appears on each JR pass for sale. Moreover, all the JR pass numbers are sorted by value in ascending order and the index of each number in the sorted set is recorded. On the other hand, a morphed image that conceals the face image of the passenger is generated and then stored on a cloud storage server. The JR pass number and the morphed image are two essential elements for later authentication. This phase is described in detail as follows and demonstrated in Figure 2.

Step 1: Passenger PG_j presents his/her passport to an ADS.

Step 2: ADS verifies the submitted passport to confirm that PG_j is eligible.

Step 3: If PG_j is an eligible passenger, ADS takes a digital photograph I_j of PG_j as the face image of PG_j. Image I_j contains PG_j's distinct facial features and is used as the source image for morphing.

Step 4: ADS uses an authorized terminal TS to generate a morphed image. First, TS retrieves a preselected reference face image I_k as the target image for morphing. It should be noticed that the same target image I_k is used for face images (as source images) of different passengers. Then, TS selects coordinates of the control pixels C_j of I_j, coordinates of the control pixels C_k of I_k and the morphing rate α_j. Finally, TS generates a morphed image I_{jk}^m using I_j and I_k as

$$I_{jk}^m = M(I_j, I_k, C_j, C_k, \alpha_j). \qquad (10)$$

Step 5: TS selects a JR pass with a printed number PID_j on it for PG_j. TS obtains the index IND_j of PID_j in the set of all sorted JR pass numbers. Then, TS sends IND_j and I_{jk}^m to cloud storage servers CSS.

Step 6: CSS chooses the IND_j^{th} vacant location L_{jk}^m in a specified space on itself to store I_{jk}^m. From this step, we can infer that the location of I_{jk}^m on CSS depends on the value of JR pass number PID_j. That is, a smaller PID_j leads to a smaller IND_j, thus a lower location of I_{jk}^m on CSS.

Step 7: TS stores C_j and α_j on the JR pass and then issues the pass to PG_j.

3.3 Passenger authentication phase

After the pass buying phase is completed, the passenger obtains a JR pass. When a JR pass holder wants to board a vehicle, an attendant at a manned ticket gate must authenticate that the holder is the true user of the JR pass. A morphed image is retrieved from CSS according to the number printed on the JR pass, and then it is used to restore a face image via de-morphing. If the restored face image is the same as that of the JR pass holder, it can be convinced that the holder is a legal passenger. The passenger authentication phase is described as follows.

Step 1: Passenger PG_j presents the JR pass that he/she holds to an ADG.

Step 2: ADG inserts the JR pass into an authorized terminal TG and TG verifies that the pass is legal. After that, TG extracts PID_j, C_j and α_j from the pass.

Step 3: TG obtains the index IND_j of PID_j and then sends IND_j to CSS.

Step 4: CSS uses IND_j to determine the location L_{jk}^m where the morphed image I_{jk}^m is stored, and then retrieves I_{jk}^m.

Step 5: CSS sends I_{jk}^m to TG.

Step 6: TG restores a face image via de-morphing. First, TG retrieves the reference image I_k that is used as the target image in previous morphing. Then, TG selects coordinates of the control pixels C_k of I_k.

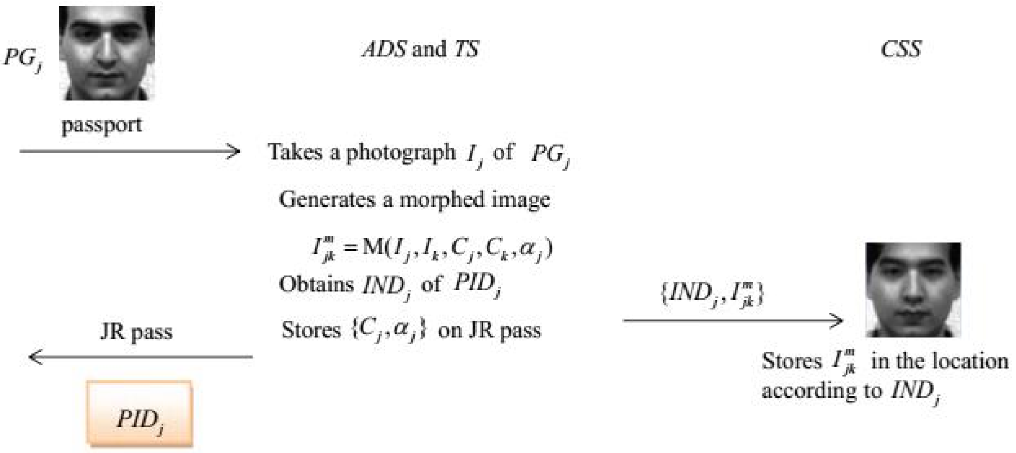

PG_j
passport
ADS and TS
CSS
Takes a photograph I_j of PG_j
Generates a morphed image
$I^m_{jk} = \mathrm{M}(I_j, I_k, C_j, C_k, \alpha_j)$
Obtains IND_j of PID_j
Stores $\{C_j, \alpha_j\}$ on JR pass
JR pass
PID_j
$\{IND_j, I^m_{jk}\}$
Stores I^m_{jk} in the location according to IND_j

Figure 2: Pass buying phase of our proposed scheme

Finally, TS restores a face image $I^{de}_{jk\to j}$ from the morphed image I^m_{jk} and the target image I_k by demorphing as

$$I^{de}_{jk\to j} = \mathrm{DM}(I^m_{jk}, I_k, C_k, C_j, \alpha_j). \qquad (11)$$

Step 7: ADG manually compares the restored face image $I^{de}_{jk\to j}$ with the face of passenger PG_j. If they are the same, PG_j is a legal passenger; otherwise, the authentication fails.

The passenger authentication phase is demonstrated in Figure 3.

4 Correctness and Security Analyses

4.1 Correctness Analysis

Correctness means that the face image of each passenger can be restored correctly in the passenger authentication phase of our proposed scheme. After the pass buying phase is completed, the passenger PG_j obtains a JR pass which has a unique, printed number PID_j and stores the coordinates of the control pixels C_j of PG_j's face image I_j and the morphing rate α_j. Meanwhile, PG_j's corresponding morphed image I^m_{jk} that conceals the information of I_j is stored on the cloud storage server CSS. Therefore, the keys for reconstructing PG_j's face image I_j are distributed among different places to enhance the security.

Let Key_1, Key_2 and Key_3 denote three different keys for reconstructing the image I_j through de-morphing. These three keys are described as follows: (1) $Key_1 = \{PID_j, C_j, \alpha_j\}$. Key_1 is stored on the JR pass held by PG_j. Among the three elements of Key_1, the JR pass number PID_j is printed explicitly on the pass. (2) $Key_2 = I_k$. Key_2 is a reference face image I_k which works together with I_j to generate a morphed image I^m_{jk}. It is secure enough to have a same Key_2 for different I_j because the combination of different I_j and a same Key_2 can generate distinguished morphed images. Thus, Key_2 can be stored on a local memory and is retrieved easily by all terminals TS and TG. (3) $Key_3 = I^m_{jk}$. Key_3 is stored on the CSS and the location where Key_3 is stored has a relationship with the value of PID_j in Key_1.

Actually, the pass buying phase is an image morphing process and the passenger authentication phase is a demorphing process. Since de-morphing is the reverse of morphing, TG can successfully reconstruct the face image of the passenger through de-morphing as long as it can obtain the same parameters used in previously conducted morphing. In other words, if TG can obtain all the keys Key_1, Key_2 and Key_3, it can ensure that face image of the passenger restored by TG is correct. As shown in Figure 4, TG can directly retrieve Key_1 from the JR pass and Key_2 from the local memory, and immediately selects C_k of I_k by using Key_2. To obtain Key_3, TG must send the index IND_j of PID_j to CSS. Then, CSS uses IND_j to determine where I^m_{jk} is stored and then transmits I^m_{jk} to TG. Consequently, TG obtains the same parameters $\{C_j, \alpha_j, I_k, C_k\}$ employed in the previous morphing and the output image I^m_{jk} of morphing.

According to the obtained parameters, TG can restore the face image of the passenger. First, horizontal-distance vector D_1 and vertical-distance vector D_2 are computed by using C_j and C_k. Then, interpolation matrices B_1 and B_2 are obtained according to D_1 and D_2. Afterwards, the warped target image I^w_k is obtained by Equation (6) and the warped source image I^w_j is computed by Equation (8). Finally, all pixels in I^w_j are shifted to their original locations in I_j by Equation (9) and the restored source image $I^{de}_{jk\to j}$ is obtained. Therefore, $I^{de}_{jk\to j}$ is the same as the face image I_j of PG_j, which implies that the correctness of our proposed scheme is achieved.

To further prove the correctness, some experiments utilizing the field morphing method [3] are conducted and the results are shown in Table 2. Assume there is a passenger PG_1 whose face image is I_1. In the pass buying phase, the morphed images with different morphing rates ($\alpha_1 = 0.1, 0.3, 0.5, 0.7$ and 0.9) can be generated from

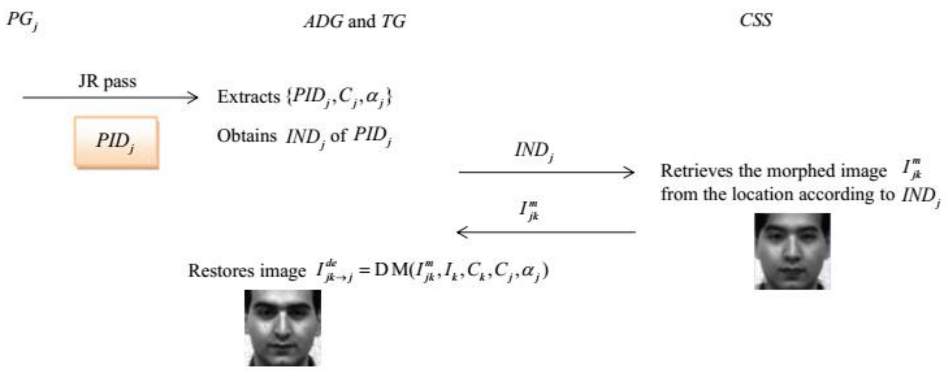

PG_j
ADG and TG
CSS
JR pass
PID_j
Extracts $\{PID_j, C_j, \alpha_j\}$
Obtains IND_j of PID_j
IND_j
Retrieves the morphed image I^m_{jk}
from the location according to IND_j
I^m_{jk}
Restores image $I^{de}_{jk\to j} = \mathrm{DM}(I^m_{jk}, I_k, C_k, C_j, \alpha_j)$

Figure 3: Passenger authentication phase of our proposed scheme

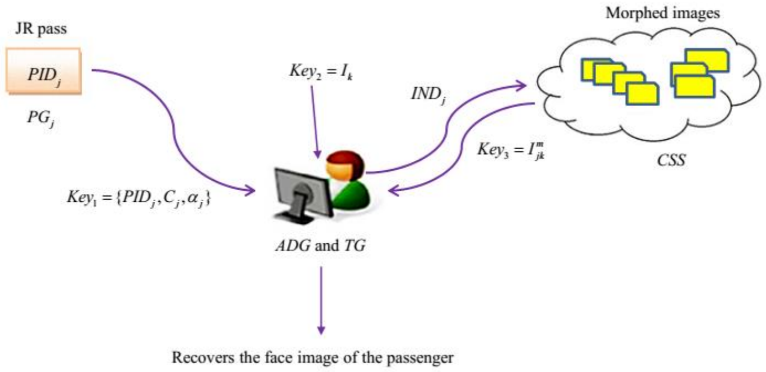
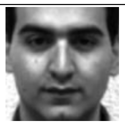
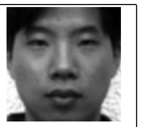
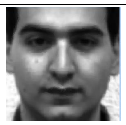
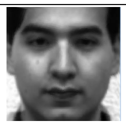
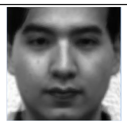
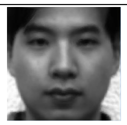
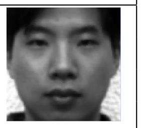
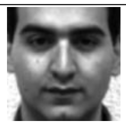
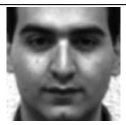
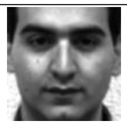
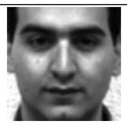
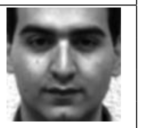


Figure 4: Keys for passenger authentication

Table 2: Experimental results

Source Image I_1 (for Passenger PG_1)			Target Image I_2		
Morphing Rate	$\alpha_1 = 0.1$	$\alpha_1 = 0.3$	$\alpha_1 = 0.5$	$\alpha_1 = 0.7$	$\alpha_1 = 0.9$
Morphed Image I^m_{12}					
Restored Image $I^{de}_{12\to 1}$					
PSNR of $I^{de}_{12\to 1}$	35.14 dB	34.83 dB	32.91 dB	31.67 dB	31.45 dB

the source image I_1 and the target/reference image I_2, as demonstrated in Table 2. It is observed that the morphing rate α_1 plays a very important role in the way the morphed image looks like for the given source image and target image. Therefore, the morphed image can be more similar to the source image when a smaller α_1 is set; on the contrary, the morphed image can be more similar to the target image when a larger α_1 is set. Table 2 also illustrates the restored images from different morphed images in the passenger authentication phase. The visual quality of the restored image is very good, thus it is impossible to visually perceive the slight difference between the original source image and the restored one.

4.2 Security Analysis

In this subsection, we analyze that our proposed scheme can enhance security significantly in terms of protecting the passenger's privacy and resisting well-known attacks.

First, the passenger's confidential information can be protected efficiently. No confidential information about the passenger but only a unique number is printed on the JR pass. This can guarantee that if the JR pass is stolen or discarded when it expires, no one can obtain the personal information from the pass. Moreover, although the face image of the passenger is used for authentication by our proposed scheme, it is not stored anywhere. Instead, a morphed image that hides the face image of the passenger is generated. To further protect the passenger's privacy, the morphed image is stored on CSS which can be retrieved only by authorized attendants in JR stations and the location for the storage of the morphed image is associated with the JR pass number. Even if an unauthorized person retrieves the morphed image from CSS, he/she is unable to know who the true passenger is since the morphed image is unlike the passenger's face image. In addition, the passport never needs to be carried with the passenger during the whole process of authentication, which extremely decreases the probability of losing the passport.

Next, we consider some attacking scenarios to analyze whether our proposed scheme is secure. Usually, a malicious attacker wants to impersonate the passenger to use the JR pass. The attacker can pass the authentication only if his/her face image can be restored by the attendant ADG who takes responsibility of the authentication. However, this impersonation attack will fail. Assume that the attacker steals or duplicates the JR pass and then presents it to an ADG. ADG conducts the operations of authentication in Subsection 3.3 and restores a face image via de-morphing. Because all the keys Key_1, Key_2 and Key_3 for de-morphing are identical to those used in morphing, ADG definitely restores the face image of the true passenger. Then, ADG can immediately distinguish the differences between the restored face image and that of the attacker. Therefore, the attacker cannot pass the identity authentication.

Since the attacker becomes aware that it is impossible for him/her to pass the authentication in the above situation, he/she tries to forge the data in Key_1 stored in the JR pass. If the attacker modifies the parameter PID_j in Key_1 and sends Key_1 to ADG, ADG will retrieve another morphed image I'^{m}_{jk} that was created from another passenger's face image, resulting in an incorrect Key_3. Then, ADG uses correct key Key_2 and incorrect keys Key_1 and Key_3 to restore a face image via de-morphing. However, the restored face image is unlike that of the attacker due to the incorrect keys so that the attack can be detected easily. The attacker may also modify the values of C_j and α_j in Key_1. Obviously, Key_1 becomes incorrect and it similarly leads to the fact that the restored face image is different from that of the attacker. Therefore, no matter how to forge the information stored in the JR pass, the attacker cannot make the face image of him/her look the same as the restored one.

Moreover, an attacker may launch an attack on ADG and obtain Key_2, *i.e*, the reference face image I_k. Then, the attacker can forge Key_3 by creating a morphed image using his/her face image and I_k. However, the attacker cannot put the forged Key_3 onto CSS since the attacker does not have the privilege of accessing CSS. Thus, ADG still retrieves correct Key_3 rather than the forged Key_3 and recovers a different face image from that of the attacker. In a word, the attacker fails to launch the impersonation attack on the true passenger due to the strategy that the keys for authentication are distributed among different places.

In the following, some experiments using the test images in Yale face database [5] and AT&T face database [4] are performed to further support the above analyses. Four attacking scenarios for impersonation attacks are considered, *i.e.*

1) The JR pass number PID_j in Key_1 is modified with the morphing rate $\alpha_j = 0.5$ while other elements in Key_1 are the same;

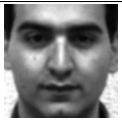
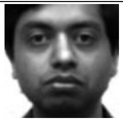
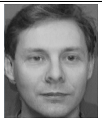
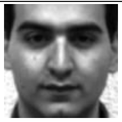
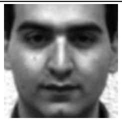
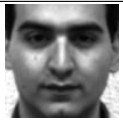
2) The coordinates of the control pixels C_j in Key_1 is modified with the morphing rate $\alpha_j = 0.5$ while other elements in Key_1 are the same;

3) α_j is modified to be 0.8 while other elements in Key_1 are the same;

4) Key_3 is forged by morphing I_k in Key_2 and the face image of the attacker I_α with the morphing rate $\alpha_j = 0.5$.

Table 3 demonstrates the attacking results on passenger PG_1 under the four scenarios mentioned above, using PSNR values to check whether the impersonation attacks are successful. From Table 3 we can see that all of the PSNR values are very small (less than 13 dB), which indicates that the difference between the restored image and the attacker's face image is very large, evaporating the attacker's attempt to restore a similar face image to that of him/her. Moreover, the PSNR values in the first and the last scenarios are extremely small (5.67 dB and 7.38 dB).

Table 3: Attack experiments

	Legal User PG_1 (Image I_1)		Attacker (Image I_α)	
	Attacking Scenario 1	Attacking Scenario 2	Attacking Scenario 3	Attacking Scenario 4
Recovered Image				
PSNR	5.67 dB	12.47 dB	12.25 dB	7.38 dB

This is because the modified parameters in these two scenarios have made more significant impact on restoration according to previous analyses in this subsection.

5 Conclusions

In this paper, we proposed a novel passenger authentication scheme for the JR pass, which innovatively adopts image morphing and cloud storage techniques to significantly enhance the security compared to traditional JR pass authentication mechanisms. The proposed scheme has the following contributions:

1) Only an assigned, unique number is printed on each JR pass to protect the passenger's privacy;

2) A morphed image is generated by the face image of the passenger and a pre-selected reference image through morphing, thereby hiding the face image of the passenger into the morphed image;

3) The generated morphed image is stored on the cloud storage, which efficiently avoids the disclosure of the passenger's personal information;

4) The authentication for the JR passenger is quite simple by performing an image de-morphing process. The proposed scheme can significantly increase both security and convenience according to our analyses.

References

[1] *Japan Railways Group. Explore with a Japan rail pass.* http://www.jrpass.com,accessed, Dec. 2016.

[2] *JRPass ltd. Japan rail pass: your sightseeing passport to Japan.* http://www.japanrailpass.net/en/index.html, Dec. 2016.

[3] T. Beier and S. Neely, "Feature-based image metamorphosis," *ACM SIGGRAPH Computer Graphics*, vol. 26, pp. 35–42, 1992.

[4] AT&T face database. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.

[5] Yale face database. http://cvc.yale.edu/projects/yalefaces/yalefaces.html.

[6] X. Huang, Y. Xiang, A. Chonka, J. Zhou, and R. H. Deng, "A generic framework for three-factor authentication: preserving security and privacy in distributed systems," *IEEE Trans. Parallel Distrib. Syst*, vol. 22, pp. 1390–1397, 2011.

[7] W. S. Juang, S. T. Chen, and H. T. Liaw, "Robust and efficient password-authenticated key agreement using smart card," *IEEE Trans. Ind. Electron*, vol. 55, pp. 2551–2556, 2008.

[8] L. Lamport, "Password authentication with insecure communication," *Commun. ACM 24*, pp. 770–772, 1981.

[9] C. T. Li, "A new password authentication and user anonymity scheme based on elliptic curve cryptography and smart card," *IET Inf. Secur*, vol. 7, pp. 3–10, 2013.

[10] X. Li, J. Niu, M. K. Khan, and J. Liao, "An enhanced smart card based remote user password authentication scheme," *J. Netw. Comput. Appl*, vol. 36, pp. 1365–1371, 2013.

[11] X. Li, J. Niu, Z. Wang, and C. Chen, "Applying biometrics to design three-factor remote user authentication scheme with key agreement," *Secur. Commun. Netw*, vol. 7, pp. 1488–1497, 2014.

[12] Q. Mao, K. Bharanitharan, and C. C. Chang, "Edge directed automatic control point selection algorithm for image morphing," *IETE Tech. Rev*, vol. 30, pp. 343–243, 2013.

[13] Q. Mao, K. Bharanitharan, and C. C. Chang, "A proxy user authentication protocol using source-based image morphing," *Comput. J*, vol. 58, pp. 1573–1584, 2015.

[14] Q. Mao, C. C. Chang, L. Harn, and S. C. Chang, "An image-based key agreement protocol using the morphing technique," *Multimed. Tool. Appl*, vol. 74, pp. 3207–3229, 2015.

[15] Q. Zhao, M. Akatsuka, and C. H. Hsieh, "Generating facial images for steganography based on iga and image morphing," in *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 364–369, Seoul, Korea, 2012.

[16] Q. Zhao and C. H. Hsieh, "Card user authentication based on generalized image morphing," in *The 3rd International Conference on Awareness Science and Technology*, pp. 117–122, Dalian, China, 2011.

Biography

Yanjun Liu received her Ph.D. degree in 2010, in School of Computer Science and Technology from University of Science and Technology of China (USTC), Hefei, China. She was an assistant professor serving in Anhui University in China from 2010 to 2015. She serves as a senior research fellow in Feng Chia University in Taiwan since 2015. Her specialties include E-Business security and electronic imaging techniques.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from February 2005. He is currently a Fellow of IEEE and a Fellow of IEE, UK. He consecutively won Outstanding Talent in Information Sciences of the R. O. C., AceR Dragon Award of the Ten Most Outstanding Talents, Outstanding Scholar Award of the R. O. C., Outstanding Engineering Professor Award of the R. O. C., Distinguished Research Awards of National Science Council of the R. O. C., Top Fifteen Scholars in Systems and Software Engineering of the Journal of Systems and Software, and so on. His current research interests include database design, computer cryptography, image compression, and data structures.