

Tamper Detection in Multimodal Biometric Authentication Systems Using Fragile Fingerprint Watermarking and Convolutional Neural Networks

Abdulmawla Najih*¹, Nooreddin Hemidat², Abier Belashher³

^{1,2}Department of computer engineering, The Higher Institute of Science & Technology Gharian, ³Department of computer engineering, Faculty of Engineering, University of Tripoli

*nabulmawla@gmail.com

الملخص

النمو السريع في توظيف المصادقة البيومترية متعدد الوسائط لحماية المعلومات والخدمات لاقت اهتمامًا كبيرًا لاستخدامها في تأمين نقاط الضعف في هذه الأنظمة. تعد العلامة المائية الرقمية إحدى التقنيات الرئيسية المستخدمة لضمان أمن هذه الأنظمة. يسمح استخدام العلامة المائية الرقمية للمصادقة البيومترية بالتعرف على الصور الأصلية التي يتم إرسالها بين الأجزاء المختلفة من النظام، أو المخزنة في قاعدة البيانات. ويمكن أيضًا استخدام العلامات المائية الهشة للكشف عن اختراق والتلاعب في الصور، بالإضافة إلى مهمة التعرف على الصور الأصلية. في هذه الدراسة، تم اقتراح تقنية العلامة المائية الرقمية الهشة، والتي تستخدم صورة بصمة الإصبع كعلامة مائية على صور الوجه. تجمع الطريقة المقترحة بين تقنية تحويل جيب التمام المنفصل (DCT) وتقنية العلامة المائية للبت الأقل أهمية (LSB) يسمح هذا المزيج بضغط الصورة ذات العلامة المائية، حيث إنه يعالج نسخة DCT لصورة الغلاف، مع الحفاظ على هشاشة العلامة المائية، باستخدام تقنية LSB. علاوة على ذلك، تقوم الطريقة أيضًا بتشفير صورة بصمة الإصبع، باستخدام Arnold Transformation، لإضافة طبقة أخرى من الأمان إلى نظام المصادقة البيومترية. وقد أظهرت نتائج التقييم أن الطريقة المقترحة قد تفوقت على أحدث الأساليب الموجودة في الدراسات السابقة، لأنها تحافظ بشكل كبير على المعلومات في كل من بصمة الإصبع والصور الحيوية الأخرى، بحيث يمكن استخدام كليهما في المصادقة العملية دون الحاجة إلى استخدامها بشكل منفصل.

الكلمات المفتاحية: المصادقة متعددة الوسائط البيومترية، علامة مائية هشة، تحويل جيب التمام منفصلة، البت الأقل أهمية، بصمات الأصابع.

Abstract

The rapid growth of multimodal biometric authentication employment to protect information and services has attracted significant attention toward securing the vulnerabilities in these systems. One of the main techniques that are used to improve the security of these systems is digital watermarking. The use of digital watermarking allows the biometric authentication to recognize the authenticity of the images communicated among the different parts of the system, or stored in a database. Fragile watermarking can also be employed for tamper detection, in addition to the authenticity recognition task. In this study, a fragile digital watermarking technique is proposed, which uses the fingerprint image as a watermark on face images. The proposed method combines the Discrete Cosine Transform (DCT) and Least Significant Bit (LSB) watermarking technique. This combination allows the compression of the watermarked image, as it manipulates the DCT version of the cover image, while maintaining the fragility of the watermarking, using the LSB technique. Moreover, the method also encrypts the fingerprint image, using Arnold Transformation, to add another layer of security to the biometric authentication system. The evaluation results show that the proposed method has outperformed the state-of-the-art methods existing in the literature, as it highly maintains the information in both the fingerprint and the other biometric images, so that, both can be used in the authentication process without the need to communicate them separately.

Keywords: Multimodal biometric authentication; fragile watermarking; Discrete Cosine Transform; Least Significant Bit; Fingerprints.

I. INTRODUCTION

Biometric authentication systems are being widely employed to protect information and services from any unauthorized access. These systems have shown better resistance to simple attacks, such as shoulder surfing, which the earlier secret-based systems suffer from [1, 2]. However, the use of biometric authentication still suffers from vulnerabilities at different positions of the system. One of the main concerns in these systems is the authenticity of the images received from the sensors or stored in the models' database [3]. Image

processing tools can be used to manipulate these images and change certain features in order to gain unauthorized access to the system [4]. To verify the authenticity of the received images, two main approaches are used, which are the cryptography based [5, 6] and fragile watermarking based approaches [7, 8]. In cryptography based approaches, a hash function is used to calculate a message authentication code, which is compared to the code calculated for the received image, using the same hash function, in order to verify the authenticity of the received image. In fragile watermarking based approaches, a watermark is inserted in the image before being transmitted, where the receiver verifies the authenticity of the image by investigating the existence of the watermark. However, the existing watermarking techniques insert static watermarks, or watermarks related to features from the image being protected, so that, the receiver can extract this information from the image and validate the watermark to detect any tampering [9].

With the growing importance and sensitivity of information and services protected by biometric authentication systems, and according to the better accuracy and wider population coverage of multibiometric authentication systems, these systems are being widely used in the recent years [10]. Such systems extract biometric features from multiple body parts of the user authenticating into the system. Many of these systems rely on extracting features from the fingerprint and face images, according to the high availability, distinctiveness and robustness of these biometrics [11, 12] and the ability of collecting the face image, passively, during the collection of the fingerprint [13]. However, in addition to the importance of the authenticity verification of the stored model images and those collected from the user, it is important to maintain the biometric features in these images as intact as possible, to maintain the accuracy of the authentication system [14]. Moreover, watermarking techniques are used to embed one of the images in the other, in order to reduce the bandwidth required by the authentication system to communicate these images [15].

A hybrid digital watermarking technique is proposed by Vatsa et al. [16] that watermarks the face image information on the fingerprint image, by combining the Discrete Wavelet Transform (DWT) and LSB techniques. To increase the efficiency of the watermark data, the two-dimensional Gabor of the face image is calculated and used as the watermark data on fingerprint images. The results of this study show that the watermarking technique has been able to survive through different geometric and frequency attacks. Thus,

the proposed bimodal authentication system has been able to significantly maintain the recognition rate, despite the application of multiple attacks. However, the use of such approach does not allow the detection of any tampering, where some information in the cover photo may be manipulated to produce a false authentication.

Thanki and Borisagar [17] propose a fragile watermarking technique that hides the fingerprint image, as the watermark, in the face image, as the cover image. This method combines the Singular Value Decomposition (SVD) to embed the details wavelet of the DWT of the fingerprint image, which is encrypted using the Compressive Sensing (CS) technique, to add another layer of protection. Although the watermark data are compressed, to reduce the distortion imposed on the cover image, no compression is applied to the cover image, which increases the storage required to store these images and the bandwidth required to communicate them. However, the fragility of this technique has enabled the detection of any tampering with the watermarked image, as the similarity measures between the original watermark and the received one is very low when the watermarked image is attacked. Similarity, the method proposed in [18] also uses the CS theory but combined with the Fast Discrete Wavelet Transform. Despite the lower effect of this method, imposed by the watermark over the cover image, the results show that compressing the image results in losing the watermark information. Hence, the watermarked images are required to be stored and communicated in full size. Such requirement increases the resources consumption of the system, i.e. the storage space and bandwidth. Moreover, the fragile watermarking technique proposed in [19] also uses Discrete Wavelet Transform (DCT) but does not consider compressing the watermarked image.

A watermarked image can be compressed when a robust watermarking technique is used. Nafea et al. [20] present a hybrid watermarking technique that uses the Discrete Wavelet Transform-Singular Value Decomposition. The results of the evaluation experiments show that the watermarked image survives compression, up to a certain rate. However, as a robust watermarking technique, tamper cannot be detected in an images watermarked using such technique [21, 22]. Hence, despite the reduction in the resources consumption when such a method is used, tampering cannot be detected, especially when an attack targets certain region of the image.

To reduce the effect of the watermark information on the cover image and reduce the size of the watermarked image, by allowing the use of smaller

cover images, the method proposed by Ma et al. [23] down samples the watermark image. The face images are used as the watermark, for the fingerprint image, and are reduced to only 8×8 pixels before being used as the watermark. Accordingly, the facial features cannot have significant influence in the authentication stage, as these features are downsized, according to the results of the conducted experiments. However, the results show that the method has been able to protect the watermarked images, as the watermark information is lost when any attack is applied to the watermarked image.

This paper proposes a hybrid fragile watermarking technique that combines Discrete Cosine Transform (DCT) and Least Significant Bit (LSB) techniques. The use of the DCT format of the image allows compressing it before being stored or communicated, which reduces the storage and bandwidth required by the authentication system, while the use of the LSB method provides the required fragility to detect any tampering in the image. JPEG image format, which is the most widely used compressed image format [24, 25], relies of DCT to eliminate cosine frequencies that have least magnitude in the image. However, the use of LSB watermarking technique prior to applying the DCT eliminates the entire watermark information during compression. Thus, the proposed method applies the LSB algorithm to the DCT values of the image, instead of the actual pixels' values. The use of the fingerprint image as a watermark in the face image reduces the size of the stored and communicated data. However, the proposed method maintains two important conditions, which are preserving the information in both images as intact as possible, and detect any tampering occurs to the resulting watermarked image. As the watermark information in the proposed method is not extracted from the cover image, tamper detection at the receiving end cannot be achieved using the traditional techniques. Thus, a Convolutional Neural Network (CNN) is trained to recognize the patterns in fingerprint images, so that, is such patterns are not detected, the image is considered unauthentic.

The remainder of this paper is organized as follows. Section II describes the proposed techniques to implement the tamper detection method, which are the watermarking and fingerprint patterns detection techniques. Section III describes the dataset used for the evaluation process and the adjustments conducted to produce more appropriate sets. Section IV describes the experimental setup and the performance measures of the proposed methods.

Section V presents the conclusions of this study and the future work that may improve the performance of the proposed methods.

II. PROPOSED METHOD

JPEG image compression standard [26] uses two compression stages to reduce the size of a still image. The first stage uses lossy compression based on calculating coefficient values of cosine waves with 64 different frequencies for each 8×8 pixels in that image. Then, the resulting values are divided by corresponding values in a predefined quantization table, which mainly reduces the effect of higher frequencies as they normally have less visual effect on the image. The resulting values are then rounded to the nearest integer, in order to reduce the number of unique values in the resulting matrix. Up to this level of JPEG compression, the size of the resulting matrix is identical to the size of the original image, as each 8×8 pixels are replaced with 8×8 values that represent the quantized cosine coefficients. The second stage executes lossless compression using Huffman encoding, which makes use of the lower number of unique values, to reduce the size of the resulting matrix before communicating or storing the image. The proposed method interacts with the quantized values, before being compressed using Huffman encoding, or being used to retrieve the original pixels values of the image.

A. DIGITAL WATERMARK EMBEDDING

The proposed method uses the fingerprint image as the watermark to be inserted to the face image. Fingerprints impressions generated by the ripples on the surface of the finger skin, where each pixel in the image can be either belong to a ripple or not. Thus, binarizing the fingerprint image does not cause loss in the biometric information that can be extracted from it [27]. However, binarization the image can significantly reduce the size of the data required to describe the fingerprint, which reduces the distortion it imposes over the cover image, which is the face image. Before binarizing the fingerprint image, the image is resized to fit into the face image, if the fingerprint image is larger than the face image, using linear interpolation method [28]. Then, the intensity histogram of the image is equalized, to improve the quality of the binarized image.

To improve the security of the proposed watermarking method, the binarized fingerprint image is scrambled using Arnold Transform. This scrambling technique requires a secret key that consists of three number. The

first two number are any pre-shared secret numbers that are used in Equation 1 to calculate the new position of a point located at (x, y) coordinates based on its previous coordinates, while the third number represents the number of iterations that the encoding is applied over the image [29, 30]. Thus, these numbers can be set to static values or time-sensitive values to deny any replay attacks

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} 1 & a \\ b & ab + 1 \end{bmatrix} \begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix} \text{mod}(D) \quad (1)$$

The scrambled version of the binarized fingerprint image is then used to manipulate the LSB of the DCT coefficient values, calculated from the face image. As the dimensions of the fingerprint image is equal to or less than the dimensions of the face image, and as each pixel value in the fingerprint image can only be zero or one, it is guaranteed that LSBs of the face image can handle the entire information of the fingerprint image. Finally, Huffman encoding is applied to the manipulated values before storing or communicating the resulting values in JPEG format, as shown in Algorithm 1.

Algorithm 1: Watermark Embedding

Input: Fingerprint image, Face image, Arnold keys

(a, b, i)

1: FL, FW \leftarrow Length and width of face image

2: PL, PW \leftarrow Length and width of fingerprint image

3: if PL > FL:

Fingerprint \leftarrow Resize(Fingerprint, FL/PL)

PL, PW \leftarrow Length and width of fingerprint image

3: if PW > FW:

Fingerprint \leftarrow Resize(Fingerprint, FW/PW)

PL, PW \leftarrow Length and width of fingerprint image

4: Fingerprint \leftarrow EqualizeHist(Fingerprint, (0,255))

5: for x = 0 to PW:

for y = 0 to PL:

if Fingerprint[x,y] > 127:

Fingerprint[x,y] \leftarrow 1

Else:

Fingerprint[x,y] \leftarrow 0

6: Fingerprint \leftarrow Arnold Transform(Fingerprint, (a, b, i))
7: FDCT \leftarrow DCT(Face)
8: FDCT \leftarrow Quantize(FDCT)
9: for x = 0 to PW:
 for y = 0 to PL:
 FDCT[x,y].LSB \leftarrow Fingerprint[x,y]
10: FDCT \leftarrow Huffman_Encode(FDCT)
Output: FDCT

B. Fingerprint and Face Images Extraction

When the compressed watermarked image is received, the fingerprint and face images are extracted. Both images are used for biometric authentication, while the fingerprint image is also used to detect any tampering with the watermark image. The first step to retrieve these images is to decompress the Huffman-encoded values. As the dimensions of the fingerprint image are constant, depending on the specification of the sensor used to collect these images, these dimensions are known to the receiver of the watermarked image. These dimensions are compared to the dimensions of the received image in order to recognize the area that contains the data of the fingerprint image. By collecting the LSBs of the received coefficient values, the original values of the scrambles binarized fingerprint image can be retrieved. Then, using the same keys used to scramble the fingerprint data, inverse Arnold transformation is used to reconstruct the original binarized fingerprint image. Finally, by applying inverse DCT to the received coefficients, the face image can also be reconstructed. Algorithm 2 describes the main steps required to retrieve the fingerprint and face images from the received compressed coefficient values.

Algorithm 1: Fingerprint and face image extraction

Input: Compressed DCT coefficients, Fingerprint length (PL) and width (PW), Arnold keys (a, b, i)
1: Coeff \leftarrow Huffman_Decode(Compressed)
2: FL, FW \leftarrow Length and width of Coeff
3: if PL>FL:
 PW \leftarrow int(FL/PW)
 PL \leftarrow FL

```
4: if PW>FW:
    PL ← int(FW/PW)
    PW ← FW
5: Fingerprint ← Empty_array(PW, PL)
6: for x = 0 to PW:
    for y = 0 to PL:
        Fingerprint[x,y] ← Coeff[x,y].LSB
7: Fingerprint ← Inverse Arnold
Transform(Fingerprint, (a, b, i))
8: Face ← Inverse DCT (Coeff)
Output: Fingerprint, Face
```

C. Tamper Detection

As the watermark images are not static or extracted from features in the cover image, and as fingerprint images are different among individuals, it is not possible to use a similarity measure between the retrieved watermark and a model image to detect any tampering with the received image. Thus, a machine learning technique, based on Convolutional Neural Network (CNN) is proposed to learn the patterns of a fingerprint. This neural network can then be used to detect fingerprint images in the extracted watermarks, so that, if such patterns are missing, the image is considered to be tamper with.

As shown in Figure 1, the implemented CNN consists of three convolutional layers, with 128, 64 and 32 filters of 2×2 pixels size, each followed by a MaxPooling layer with 2×2 -pixel filter size, to maintain accurate positioning of the detected patterns. These layers are followed by two hidden fully-connected layers, with 256 and 128 neurons with 50% dropout rate each, to avoid overfitting. The output layer consists of a single neuron, as only one output value is required, which represents the probability of the input containing a fingerprint image. All layers, except the output layer use Rectified Linear Unit (ReLU) activation function, according to the good performance and fast learning rates they provide. The sigmoid activation function is used for the neuron in the output layer, as the output is required to be in the range $[0,1]$. The CNN is trained using a set of images, where images that contain fingerprint images are labeled with one, while those that do not have fingerprint images are labeled with zero.

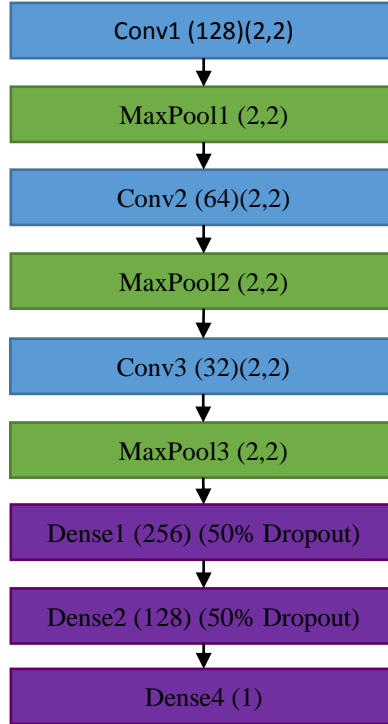


Figure 1: Structure of the fingerprint detection CNN.

III. FINGERPRINTS AND FACE IMAGES DATASETS

To evaluate the performance of the proposed methods, two fingerprint and three face images datasets are used to measure the similarity between the cover images, before and after watermarking, as well as the detection rate of the CNN. The FVC 2002DB1 and FVC 2004DB4[31] fingerprint dataset are used, as these datasets are used in earlier studies. For face images, the Indian Faces[32], the ORL Database of Faces and the FERET[33] Faces dataset are used. These datasets are selected to allow comparison between the proposed method and the state-of-the-art methods that exist in the literature. Table summarizes the contents of each of the datasets used in this study. The ORL and FERET face images dataset are combines with the fingerprint images of the FVC 2002 dataset, while the Indian Faces dataset is combined with the fingerprint images from the FVC 2002 and FVC 2004.

IV. EXPERIMENTAL SETUP AND RESULTS

All experiments are implemented using Python programming language [34], where the OpenCV library [35] is used for image processing while Keras artificial neural networks library [36] is used with Tensorflow [37] machine learning library as its backbone. The SciKit-Learn [38] library is used to calculate the performance measures of the proposed method. These experiments are conducted using an Intel® Core™ i7-7700HQ processor at 2.81GHz frequency and 16.0GB of memory running with Windows 10 Pro operating system. A GTX1080Ti Graphical Processing Unit (GPU) is used to accelerate the computations required by the CNN in the tamper detection phase.

A. EXPERIMENTAL METRICS

The performance of the proposed method can be illustrated using four aspects:

- 1) The distortion in the fingerprint and face images imposed by the watermarking process, as these images are required in the authentication process.
- 2) The tamper detection accuracy under different attacks executed against the watermarked image.
- 3) The reduction in the size of the data required to be transferred, when the watermarked image is used instead of communicating both images solely.
- 4) The complexity of the proposed method, measured by computing the time consumed by the proposed method to watermark the images, extract the original images and detect any tampering with the watermarked image.

The distortion imposed by the watermarking can be illustrated by measuring the similarity between the fingerprint image before watermarking and those extracted from the received watermarked image, and the face images before and after the fingerprint images are inserted in them. Two similarity measures are used to calculate these similarities, which are the Peak Signal to Noise Ratio (PSNR) [39] and the Structural Similarity Index Measure (SSIM) [40]. Equation (1) is used to calculate the PSNR, while the SSIM is measured using Equation (2).

$$PSNR = 20 \log_{10} Max - 10 \log_{10} MSE, \quad (1)$$

where,

$$MSE = \frac{1}{m \times n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - R(i,j)]^2$$

$$SSIM = \frac{4\sigma_{IR}\bar{I}\bar{R}}{(\sigma_I^2 + \sigma_R^2)[(\bar{I})^2 + (\bar{R})^2]} \quad (2)$$

where,

$$\bar{I} = \frac{1}{n} \sum_{i=1}^n I_i, \quad \bar{R} = \frac{1}{n} \sum_{i=1}^n R_i,$$

$$\begin{aligned} \sigma_I^2 &= \frac{1}{n-1} \sum_{i=1}^n (I_i - \bar{I})^2, & \sigma_R^2 \\ &= \frac{1}{n-1} \sum_{i=1}^n (R_i - \bar{R})^2, \end{aligned}$$

$$\sigma_{IR} = \frac{1}{n-1} \sum_{i=1}^n (I_i - \bar{I})(R_i - \bar{R}).$$

The tamper detection accuracy is measured by calculating the False Acceptance Rate (FAR), which represents the ratio of tampered images predicted as normal, and False Rejection Rate (FRR), which represents the ratio of untampered images predicted to be tampered with. These measures are calculated using Equation (3) and (4), respectively.

$$FAR = \frac{FP}{FP + TN} \quad (3)$$

$$FRR = \frac{FN}{TP + FN} \quad (4)$$

where,

TP: Untampered images predicted as untampered.

TN: Tampered images predicted as tampered.

FP: Tampered images predicted as untampered.

FN: Untampered images predicted as tampered.

B. EXPERIMENTS ASSUMPTION

In the conducted experiments, the face images in the selected dataset have dimensions of 119×92 pixels, while the fingerprint images have 200×200 pixels. All JPEG compressions are set to 90%, in order to maintain acceptable quality for the biometric authentication stage. The keys used for Arnold transformation are $a=1$, $b=1$, $i=10$.

C. FINGERPRINT AND FACE IMAGES SIMILARITY

The block diagram shown in Figure 3 illustrates the procedure followed to measure the similarity between fingerprint and face images. Per each individual in the dataset used for evaluation, the average PSNR and SSIM values are calculated, which are illustrated in Table

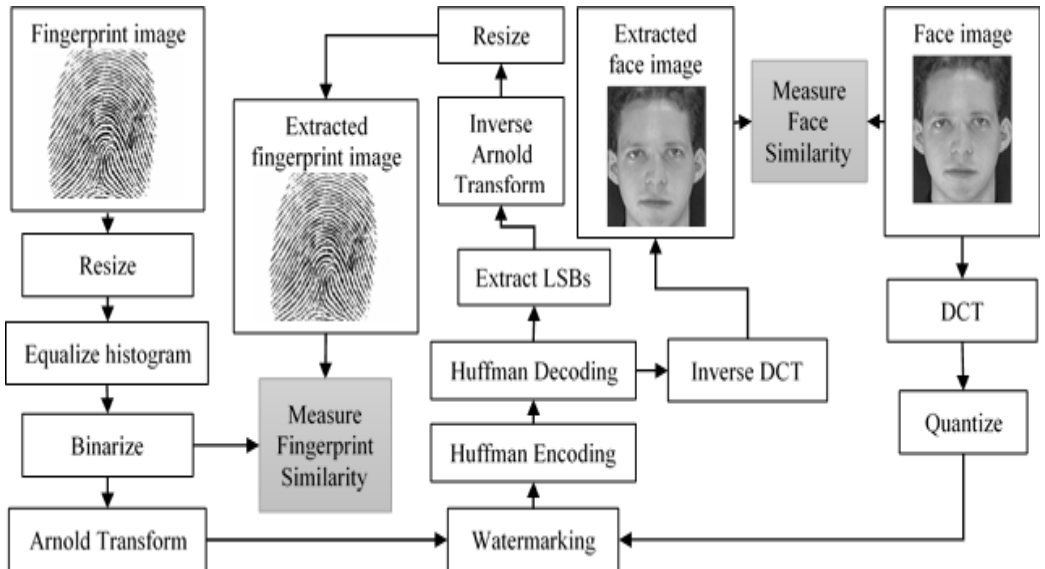


Figure 2: Fingerprint and face images similarity measurement block diagram

Individual	Face Images		Fingerprint Images	
	Average PSNR (dB)	Average SSIM	Average PSNR (dB)	Average SSIM
1	46.71	0.9983	58.23	0.9977
2	49.15	0.9994	57.42	0.9971
3	48.14	0.9988	56.86	0.9967
4	45.06	0.9967	56.77	0.9967
5	41.6	0.994	57.08	0.997
6	42.36	0.9966	57.26	0.9971
7	48.17	0.9992	57.29	0.9971
8	47.89	0.9984	57.07	0.9969
9	35.38	0.9886	56.96	0.9968
10	49.1	0.9993	57.23	0.997
11	47.91	0.9991	57.24	0.9971
12	39.79	0.9933	58.09	0.9977
13	45.98	0.9982	57.3	0.9971
14	45.69	0.9981	58.36	0.9975
15	49.12	0.9994	57.03	0.9969
16	49.18	0.9994	57.09	0.9969
17	45.26	0.9987	57.16	0.997
18	40.83	0.9926	57.67	0.9975
19	43.17	0.9975	57.08	0.997
20	48.24	0.9991	57.35	0.9972
21	43.17	0.9966	57.33	0.9971
22	49.12	0.9995	57.75	0.9974

23	45.06	0.9975	57.2	0.9971
24	44.29	0.9992	57.28	0.997
25	37.51	0.9915	57.22	0.997
26	36.99	0.9905	57.55	0.9973
27	47.83	0.9991	56.94	0.9969
28	45.86	0.9983	57.08	0.997
29	45.33	0.9985	56.76	0.9967
30	47.96	0.9993	57.73	0.9974
31	44.25	0.9987	58.15	0.9976
32	41.1	0.9959	57.37	0.9972
33	49.18	0.9995	56.66	0.9966
34	41.37	0.9993	56.8	0.9969
35	42.02	0.9944	57.22	0.9971
36	49.06	0.9994	57.44	0.9972
37	48.13	0.9988	57.04	0.9969
38	47.26	0.9982	57.3	0.9969
39	49.2	0.9994	56.31	0.9963
40	43.86	0.9959	57.38	0.9973
Overall Average:	45.18	0.9974	57.44	0.9973

The results of this experiment show that the similarity between the face images, before and after being watermarked, vary from 35.38 to 49.20 dB PSNR with an average of 45.18, while the similarity measures vary from 99.86% to 99.95%, with an average of 99.74%. Moreover, the similarity between the binarized fingerprint image prior to the watermarking and the image extracted from the watermarked image varies from 56.31dB to 58.36dB PSNR, with an average of 57.44, while the SSIM measure varies

from 99.63% to 99.77%, with an average of 99.73%. These results show that the distortion imposed by the watermarking technique is so low that it does not affect the features in the images, which are used for biometric matching. Moreover, binarizing the fingerprint image has been able to increase the similarity measures of the fingerprint images, and reduce the distortion in the face images, as less data are watermarked in them.

In comparison, the method presented by Noore et al. [14] watermarks the face image over the fingerprint image for multimodal biometric systems. This method embeds the face image in certain texture regions in the DWT of the fingerprint image, selected based on a number that is used as a secret key to decrypt the watermark data. The similarity measures of this method are 97.58%, between the original and watermarked watermark face images, and 92.59% between the cover fingerprint images, before and after adding the watermarks. These results show that the proposed method has better similarity measures, which produces more accurate biometric authentications, as the less distortion is imposed by the proposed method. This comparison also shows the use of the entire image, in the proposed method, instead of certain regions can reduce the distortion, as the density of the watermark data is reduced when the entire image is used. More comparisons to more recent techniques are shown in Table 2 to illustrate the performance of the proposed method, regarding the average PSNR and SSIM measures.

Table 2: PSNR and SSIM comparisons with the literature

<i>Study</i>	<i>PSNR (dB)</i>	<i>SSIM</i>
<i>This study</i>	<i>51.31</i>	<i>0.99735</i>
<i>Rohit et al.[18]</i>	<i>42.59</i>	<i>0.987</i>
<i>Rohit et al.[19]</i>	<i>43.62</i>	<i>0.9850</i>
<i>Nafea et al.[20]</i>	<i>23.17</i>	<i>0.9495</i>
<i>Naima et al.[22]</i>	<i>30.00</i>	<i>0.989</i>
<i>Rohit et al.[17]</i>	<i>44.52</i>	<i>0.4998</i>

C. TAMPER DETECTION ACCURACY

In this experiment, the fragility of the proposed watermarking technique is evaluated, alongside with the temper detection rate, using the trained CNN. First, the performance of the trained CNN is evaluated when no attack is executed against the watermarked image. Then, different attacks are executed before extracting the watermark from the image and forward it to the tamper detection method. In both

cases, the detection rate is measured, and a sample of the extracted watermarks is collected, to illustrate the performance of the tamper detection method and the fragility of the watermarking technique. Figure 4 shows the procedure followed during this experiment.

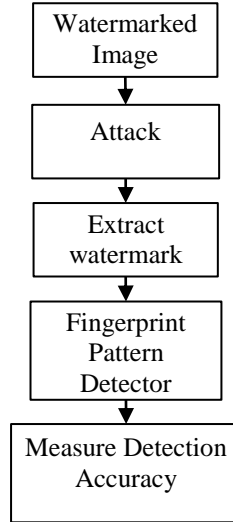






Figure 3: Tamper detection evaluation procedure.

Although the proposed watermarking method is aimed to watermark compressed images, compressing the watermarked image is one of the attacks that may be executed against it. Thus, it is important to evaluate the ability of the proposed method to detect such tampering. Moreover, attacks such as Salt & Pepper, Medial Filters and Gaussian Noise are widely used to evaluate the robustness or fragility of a watermarking technique. Thus, each of these attacks is executed on the watermarked image, before extracting the watermark information from it, using different intensities. The results of this experiment are summarized in Table 3.

Table 3: Tamper detection performance evaluation.

Attack Type	FAR	RR	Accuracy	Extracted Watermark
None	0	0	1	
Salt & Pepper (0.001)	0	0	1	
Salt & Pepper (0.002)	0	0	1	
Salt & Pepper (0.01)	0	0	1	
Salt & Pepper (0.05)	0	0	1	
Salt & Pepper (0.1)	0	0	1	
Median Filter (3x3)	0	0	1	
Median Filter (5x5)	0	0	1	
Median Filter (7x7)	0	0	1	
Median Filter (9x9)	0	0	1	
Median Filter (15x15)	0	0	1	
Gaussian Noise (0,0.01)	0	0	1	
Gaussian Noise (0,0.05)	0	0	1	
Gaussian Noise (0,0.1)	0	0	1	
Gaussian Noise (0.01,0)	0	0	1	
Gaussian Noise (0.02,0)	0	0	1	
Gaussian Noise (0.1,0)	0	0	1	
Gaussian Noise (0.05,0)	0	0	1	

JPEG compression (10%)	0	0	1	
JPEG compression (30%)	0	0	1	
JPEG compression (50%)	0	0	1	
JPEG compression (70%)	0	0	1	
JPEG compression (90%)	0	0	1	
Average:	0	0	1	

The results illustrate the perfect performance of the tamper detection method, which is a result of the combination between the fragility of the watermarking technique and the accuracy of the fingerprint pattern detection method. Thus, this combination has shown the best possible performance in detecting any type of attacks that can be executed on the biometric images, stored in the models database or communicated among the different parts of the multimodal biometric authentication system. Moreover, the use of a time-sensitive encryption key with the Arnold Transformation can protect the system from any replay attacks, where messages are intercepted and replayed to the system. The images collected from watermarks in attacked images show that the LSBs are reset after each attack is executed, which creates what appears to be a random distribution of ones and zeros.

D. IMAGES SIZE REDUCTION

As the aim of the proposed method is to protect face images using the fingerprint images while reducing the size of the watermarked image, to reduce the resources consumption, this experiment evaluates the size of the resulting image. According to the JPEG standard, a quantization table is used to reduce the number of unique frequency magnitudes, so that, the size of the resulting file required to store the image is significantly reduced when Huffman Encoding is used to store those values. However, as the proposed method adjust the value of the LSBs of the values resulting from the quantization step, the resulting file is expected to be larger than the

compressed one, without watermarking, but still smaller than the original watermarked image. At a compression rate of 90%, the proposed method has produced files with 63.87% of the original, uncompressed, file size, whereas the standard JPEG compression has produced files of 53.74% of the original size. In contrast, the methods proposed in [17-19] and [23] still require storing the file using its 100% size, while the method proposed in [20] can produce files with compression rate depending on the required file size but cannot be used for tamper detection.

E. SPEED ANALYSIS

The time consumed to execute an algorithm has significant importance when the algorithm is used with real-time applications. To evaluate the effect of the dimensions of fingerprint and face images on the time required to gen

erate the watermarked image and extract them back, these images are scaled from 20% up to 200% their actual dimensions, with a 20% step size.

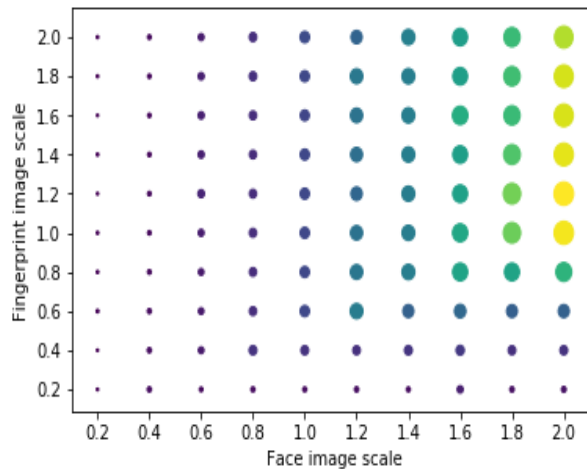


Figure 4: Scale of the fingerprint and face images.

Per each scale of the face image, all possible scales of the fingerprint images are evaluated. Then, the results are summarized, in Figure 4, per each scale of the fingerprint and face images. The minimum time consumed by the proposed watermarking algorithm, to watermark and extract the images, is 0.45mS, while the maximum time is 39mS. Moreover, the graph show that the

execution time is increased only when both the fingerprint and face images have larger dimensions.

V. CONCLUSION

This paper presents a fragile watermarking technique that combines the LSB watermarking method and DCT-based compression, to protect images communicated by multimodal biometric authentication systems. The proposed method watermarks the fingerprint image over the face image, so that, any tampering with the image causes the loss of the watermark. The use of the fingerprint image as the watermark has been able to reduce the size of the data communicated in the authentication system, as both images are transferred in a single watermarked image. However, the use of a dynamic watermark that is not extracted from features in the cover image imposes the challenge of detecting tamper in the received image. Thus, a convolutional neural network is trained to recognize patterns in fingerprint images, so that, is the watermark extracted from the received image does not have such patterns, the image is considered to be tampered with. The proposed method has shown perfect tamper detection rate, with very high similarity between the fingerprint and face images before being watermarked and after being extracted from the watermarked image and significant reduction in the size of the communicated data. The method has also shown very low time consumption in embedding the fingerprint image in the face image and extracting these images upon arrival, where the execution time has increased only when the dimensions of both images are increased.

In future work, the distortion imposed by the fingerprint watermark on other biometric images, such as iris, is going to be evaluated. As iris images are of smaller size and sharper details, compared to face image, the watermarking technique may impose a larger distortion on such images. However, the literature shows that the use of fingerprint and iris biometric is rarely combined together, as both of them cannot be collected passively.

REFERENCE

- [1] S. A. Chaudhry, K. Mahmood, H. Naqvi, and M. K. Khan, "An improved and secure biometric authentication scheme for telecare medicine information systems based on elliptic curve cryptography," *Journal of Medical Systems*, vol. 39, no. 11, p. 175, 2015.

- [2] M. Nagatomo, Y. Kita, K. Aburada, N. Okazaki, and M. Park, "Implementation and user testing of personal authentication having shoulder surfing resistance with mouse operations," *IEICE Communications Express*, vol. 7, no. 3, pp. 77-82, 2018.
- [3] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy in biometrics-based authentication systems," *IBM systems Journal*, vol. 40, no. 3, pp. 614-634, 2001.
- [4] M. Sajjad *et al.*, "Mobile-cloud assisted framework for selective encryption of medical images with steganography for resource-constrained devices," *Multimedia Tools and Applications*, vol. 76, no. 3, pp. 3519-3536, 2017.
- [5] R. Hamza, K. Muhammad, Z. Lv, and F. Titouna, "Secure video summarization framework for personalized wireless capsule endoscopy," *Pervasive and Mobile Computing*, vol. 41, pp. 436-450, 2017.
- [6] K. Muhammad, M. Sajjad, I. Mehmood, S. Rho, and S. W. Baik, "Image steganography using uncorrelated color space and its application for security of visual contents in online social networks," *Future Generation Computer Systems*, 2016.
- [7] M. M. Yeung and F. Mintzer, "An invisible watermarking technique for image verification," in *Image Processing, 1997. Proceedings., International Conference on*, 1997, vol. 2, pp. 680-683: IEEE.
- [8] N. Li, W. Du, and D. Boneh, "Oblivious signature-based envelope," *Distributed Computing*, vol. 17, no. 4, pp. 293-302, 2005.
- [9] A. Shehab *et al.*, "Secure and robust fragile watermarking scheme for medical images," *IEEE Access*, vol. 6, pp. 10269-10278, 2018.
- [10] R. Snelick, U. Uludag, A. Mink, M. Indovina, and A. Jain, "Large-scale evaluation of multimodal biometric authentication using state-of-the-art systems," *IEEE transactions on pattern analysis and machine intelligence*, vol. 27, no. 3, pp. 450-455, 2005.
- [11] J. Wayman, A. Jain, D. Maltoni, and D. Maio, "An introduction to biometric authentication systems," in *Biometric Systems*: Springer, 2005, pp. 1-20.
- [12] M. O. Oloyede and G. P. J. I. A. Hancke, "Unimodal and multimodal biometric sensing systems: a review," vol. 4, pp. 7532-7555, 2016.
- [13] T. Sim, S. Zhang, R. Janakiraman, and S. Kumar, "Continuous verification using multimodal biometrics," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 4, pp. 687-700, 2007.
- [14] A. Noore, R. Singh, M. Vatsa, and M. M. Houck, "Enhancing security of fingerprints through contextual biometric watermarking," *Forensic Science International*, vol. 169, no. 2-3, pp. 188-194, 2007.
- [15] T. Hoang, D. Tran, and D. Sharma, "Remote multimodal biometric authentication using bit priority-based fragile watermarking," in *Pattern*

- Recognition, 2008. ICPR 2008. 19th International Conference on, 2008, pp. 1-4: IEEE.*
- [16] M. Vatsa, R. Singh, A. Noore, M. M. Houck, and K. Morris, "Robust biometric image watermarking for fingerprint and face template protection," *IEICE Electronics Express*, vol. 3, no. 2, pp. 23-28, 2006.
- [17] R. Thanki and K. Borisagar, "Multibiometric Template Security Using CS Theory–SVD Based Fragile Watermarking Technique," *WSEAS Transactions on Information Science and Applications*, vol. 12, pp. 1-10, 2015.
- [18] R. Thanki and K. Borisagar, "Biometric watermarking technique based on cs theory and fast discrete curvelet transform for face and fingerprint protection," in *Advances in signal processing and intelligent recognition systems: Springer*, 2016, pp. 133-144.
- [19] R. Thanki and K. Borisagar, "Biometric Image Protection Using Compressive Sensing and DCT based Watermarking Technique," in *proceedings of RK University's First International Conference on Research & Entrepreneurship (ICRE–2016)*, 2016, pp. 1239-1248.
- [20] O. Nafea, S. Ghouzali, W. Abdul, and E.-u.-H. Qazi, "Hybrid multi-biometric template protection using watermarking," *The Computer Journal*, vol. 59, no. 9, pp. 1392-1407, 2016.
- [21] J. Hämmerle-Uhl, K. Raab, and A. Uhl, "Watermarking as a means to enhance biometric systems: A critical survey," in *International Workshop on Information Hiding*, 2011, pp. 238-254: Springer.
- [22] N. Bousnina, S. Ghouzali, M. Mikram, and W. Abdul, "DTCWT-DCT watermarking method for multimodal biometric authentication," in *Proceedings of the 2nd International Conference on Networking, Information Systems & Security*, 2019, p. 75: ACM.
- [23] B. Ma, Y. Wang, C. Li, Z. Zhang, and D. Huang, "Secure multimodal biometric authentication with wavelet quantization based fingerprint watermarking," *Multimedia tools and applications*, vol. 72, no. 1, pp. 637-666, 2014.
- [24] J.-F. Mao *et al.*, "Research on watermarking payload under the condition of keeping JPEG image transparency," *Multimedia Tools and Applications*, vol. 76, no. 6, pp. 8423-8448, 2017.
- [25] L. Dong, Q. Yan, Y. Lv, and S. Deng, "Full band watermarking in DCT domain with Weibull model," *Multimedia Tools and Applications*, vol. 76, no. 2, pp. 1983-2000, 2017.
- [26] G. K. Wallace, "The JPEG still picture compression standard," *IEEE transactions on consumer electronics*, vol. 38, no. 1, pp. xviii-xxxiv, 1992.
- [27] S. Bayram, H. T. Sencar, and N. Memon, "Efficient sensor fingerprint matching through fingerprint binarization," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 4, pp. 1404-1413, 2012.

- [28] C. De Boor, C. De Boor, E.-U. Mathématicien, C. De Boor, and C. De Boor, *A practical guide to splines*. Springer-Verlag New York, 1978.
- [29] L. Wu, J. Zhang, W. Deng, and D. He, "Arnold transformation algorithm and anti-Arnold transformation algorithm," in *Information Science and Engineering (ICISE), 2009 1st International Conference on*, 2009, pp. 1164-1167: IEEE.
- [30] A. M. Najih, S. A. R. Al-Haddad, A. R. Ramli, and S. J. Hashim, "A New Colour Image Watermarking Technique Using Special Domain," in *2015 5th International Conference on IT Convergence and Security (ICITCS)*, 2015, pp. 1-5: IEEE.
- [31] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of fingerprint recognition*. Springer Science & Business Media, 2009.
- [32] V. J. h. v.-w. c. u. e. v. I. Jain, "" The indian face database," 2002," 2002.
- [33] P. J. Phillips, H. Moon, P. Rauss, and S. A. Rizvi, "The FERET evaluation methodology for face-recognition algorithms," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 137-143: IEEE.
- [34] M. F. Sanner, "Python: a programming language for software integration and development," *J Mol Graph Model*, vol. 17, no. 1, pp. 57-61, 1999.
- [35] G. Bradski and A. Kaehler, "OpenCV," *Dr. Dobb's journal of software tools*, vol. 3, 2000.
- [36] F. Chollet, "Keras: The python deep learning library," *Astrophysics Source Code Library*, 2018.
- [37] M. Abadi *et al.*, "Tensorflow: a system for large-scale machine learning," in *OSDI*, 2016, vol. 16, pp. 265-283.
- [38] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825-2830, 2011.
- [39] C. Jinimole and A. Harsha, "Comparative Study of Different Enhancement Techniques for Computed Tomography Images," *World Academy of Science, Engineering and Technology, International Journal of Medical, Health, Biomedical, Bioengineering and Pharmaceutical Engineering*, vol. 11, no. 9, pp. 524-527, 2017.
- [40] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE signal processing letters*, vol. 9, no. 3, pp. 81-84, 2002.