Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.035830
Article

Tech Science Press

Innovative Hetero-Associative Memory Encoder (HAMTE) for Palmprint Template Protection

Eslam Hamouda[1], Mohamed Ezz[1,*], Ayman Mohamed Mostafa[1], Murtada K. Elbashir[1], Meshrif Alruily[1] and Mayada Tarek[2]

[1]College of Computer and Information Sciences, Jouf University, Sakaka, 72314, Saudi Arabia
[2]College of Computers and Informatics, Mansoura University, 35516, Egypt
*Corresponding Author: Mohamed Ezz. Email: maismail@ju.edu.sa
Received: 06 September 2022; Accepted: 28 October 2022

Abstract: Many types of research focus on utilizing Palmprint recognition in user identification and authentication. The Palmprint is one of biometric authentication (something you are) invariable during a person's life and needs careful protection during enrollment into different biometric authentication systems. Accuracy and irreversibility are critical requirements for securing the Palmprint template during enrollment and verification. This paper proposes an innovative HAMTE neural network model that contains Hetero-Associative Memory for Palmprint template translation and projection using matrix multiplication and dot product multiplication. A HAMTE-Siamese network is constructed, which accepts two Palmprint templates and predicts whether these two templates belong to the same user or different users. The HAMTE is generated for each user during the enrollment phase, which is responsible for generating a secure template for the enrolled user. The proposed network secures the person's Palmprint template by translating it into an irreversible template (different features space). It can be stored safely in a trusted/untrusted third-party authentication system that protects the original person's template from being stolen. Experimental results are conducted on the CASIA database, where the proposed network achieved accuracy close to the original accuracy for the unprotected Palmprint templates. The recognition accuracy deviated by around 3%, and the equal error rate (EER) by approximately 0.02 compared to the original data, with appropriate performance (approximately 13 ms) while preserving the irreversibility property of the secure template. Moreover, the brute-force attack has been analyzed under the new Palmprint protection scheme.

Keywords: Palmprint recognition; hetero-associative memory; neural network and Siamese network

1 Introduction

Biometric authentication is used for identifying users based on their biometric features. Traditional methods, such as security tokens and passwords, are applied to verify and authenticate users. These methods are vulnerable to brute-force and spoofing attacks. The recognition of biometric technology has

been increasing in the last few years due to the high-security features provided for identifying users. Palmprint recognition is considered an innovative technology that can achieve high accuracy and performance for user identification. The Palmprint contains sensitive security features that can be applied in civilian and computer forensics [1]. If the Palmprint of any user is stolen, he will be vulnerable his whole life. Therefore, it is sensitive information and needs to be secured. The Palmprint features generated from the Palmprint image inherit the same user sensitivity and need to be secured when stored in any authentication system. Users need to join different organizations using their Palmprint for identification and authentication. If the user uses the same template in various organizations, it will be vulnerable to cross-matching attacks. In case of one authentication system is compromised, this means that the other systems have also been compromised. Therefore, the Palmprint templates for the same user must be different and independent when enrolled in different authentication systems.

This research aims to secure the person's Palmprint feature maps (i.e., template) by translating them into an irreversible template (different feature spaces) which can be stored safely to protect the original person template from being stolen. Moreover, generating different templates from the same Palmprint enables users securely enroll in various organizations. The contributions of this research can be summarized as follows:

- Proposing a user authentication system that is based on a secure Palmprint template instead of the original Palmprint template.
- Achieving the balance between authentication accuracy (high accuracy) and the secure template complexity (irreversible template).
- The ability to generate different secured templates for the same user when registering in different authentication systems.
- Achieving the model generalization by recognizing the user (using similarity) seen during the training stage (work with unseen users).

The organization of the paper sections is explained as follows: Section 2 summarizes the relevant research works. Section 3 describes the proposed Hetero-Associative Memory encoder (HAMTE) for Palmprint template protection. Section 4 introduces the experiments, performance evaluation, and discussion. Finally, the paper is concluded in Section 5.

2 Related Works

Privacy-preserving and user authentication are considered significant security features in most recent applications. Most systems' authentication and verification processes depend mainly on a set of methods that require intensive applications for handling and verifying users. Biometric authentication is considered a novel method to verify authentic users and prevent any leakage of users' data. As presented in [2], a multimodal biometric authentication system is designed for applying security services on mobile edge computing (MEC). The research aims to detect attacks using an edge server that stores the user's secret parameters to outsource a cloud biometric database.

Different biometric authentication methods are applied, especially for fingerprint and Palmprint authentications. As presented in [3], an offline fingerprint model is proposed for predicting indoor user positions using machine learning. Large datasets are applied along with WLAN fingerprint-based building location database for training and testing samples. A fingerprint agreement method is presented in [4] for processing users' data with high-security measures. The proposed architecture applied a cloud data center as an infrastructure for maintaining secrecy and confidentiality on the cloud. The fingerprint agreement is applied on the cloud using Kerberos authentication to improve the authentication process. Another biometric authentication model on cloud computing is presented in [5]. The research aims to predict user identity by matching the feature extraction of user data with the original user features stored in a database.

An enhanced encryption technique for face authentication is proposed in [6], where a cancellable template is applied to store the authentication parameters in the database rather than storing them in biometric data. Storing cancellable templates in the database is a focal point where the templates are kept private. The template will be canceled and regenerated from the user's biometric data if the database is breached. In addition, the non-invertibility method is applied where the original biometric data will not be revoked from the transformed data, with the ability to produce different transformed versions from the user biometric data.

Other cancelable biometric systems are presented in [7,8]. The authors of [7] provided a cancelable method for speech recognition based on encryption algorithms. The user speech signal is converted into a spectrogram encrypted using optical encryption algorithms. Optical scanning holography (OSH) is applied to encrypt the speech signal and convert it into an electrical signal. An additional encryption algorithm is implemented using double random phase encoding (DRPE) to increase the security level of the encrypted signals. The proposed methodology was improved in [8] to protect the privacy of biometric data using cancelable biometrics where the data cannot be inverted so that the user information is kept private and secured. The authors suggested a hybrid algorithm based on the Fourier transform and Jigsaw transform. Two datasets are examined based on the hybrid algorithm, where the correlation scores achieved high results reflecting the proposed model's efficiency.

Palmprint technology automatically identifies and authenticates users based on their unique features. The lines in the Palmprint are considered the most significant features that can verify the users. As presented in [9], the palm lines are detected by extracting the lines in different directions. A matching score is applied between the original and the stored Palmprints to match the palm lines. The recognition of Palmprints depends mainly on the features of Palmprint images.

As shown in [10], a non-invertible Palmprint template is applied for storing geometric information. The research aims to perform a preprocessing step on the Palmprint images and then perform a feature extraction to estimate the Palmprint orientation from the template. A novel method for applying Palmprint authentication is presented in [11], where Palmprint recognition is used on mobile phones due to their extensive use. The paper's authors merge mobile face net and circle loss to build a model that can increase the accuracy of Palmprint datasets and reduce the error rate. Another novel method for Palmprint recognition is presented in [12], where SURF features are used by classifying nearest neighbor ratio and distance measure on the matched SURF key points. Three Palmprint datasets are used, namely; GPDS, IITD, and CASIA, for measuring the performance of the proposed model. For applying feature extraction on images, different categories must be identified to classify Palmprint images. These categories are image dimension, image resolution, and image acquisition.

Regarding the dimension of Palmprint images, two-dimensional (2D) and three-dimensional (3D) images are applied for verifying users. Recent research methodologies applied 3D dimensional images for higher accuracy and user verification. As presented in [13], an overall overview is applied for extracting features on 3D Palmprint data. The data is acquired and preprocessed to analyze the feature descriptors of Palmprints. As shown in [14], a survey is proposed to identify left and right images of Palmprints. 2D images are extracted, and the similarity between both left and right palms is defined. A matching score is determined to measure the relationship between the detected Palmprints and whether they are matched or not. As presented in [15], a comprehensive study has been conducted to revise the main models and methodologies applied to Palmprint technology. The authors explained that the primary strategy for using Palmprint is through the Palmprint lines, where multimodal biometrics can be applied to improve the recognition rate. The authors stated that acquiring the Palmprint images, whether in 2D or 3D, must be done through different algorithms to be stored on a specified database. The authors of [16] proposed a framework for enhancing Palmprint recognition. The framework represents discriminative features based

on multiple scenarios using deep learning convolutional networks. A deep learning algorithm is applied to extract discriminative features of the Palmprint images that can be used for verification and identification. As presented in [17], the verification of Palmprint images can be enhanced using multispectral images that can increase reliability and efficiency. Multispectral recognition can be applied through different approaches such as structured, appearance, statistical, coding, and hybrid methods. A comparative analysis is proposed to compare the different techniques and determine the feature extraction method and the type of database.

Regarding the resolution of Palmprint images, low-resolution images can contain visible lines. Still, they may have some wrinkles, while high-resolution images have clear points that can enhance the verification of Palmprint images. As shown in [18], a framework is proposed to classify the Palmprint images, whether they are 3D or high-resolution images. Regarding the acquisition of Palmprint images, recent works in Palmprint are based on capturing Palmprints without any restrictions where the Palmprint images are detected on different scales that can produce noises and losses in the Palmprint features that can affect the overall performance of verifying users. As presented in [19], additional features on Palmprint are applied to improve the accuracy of recognizing the palm features. The scale-invariant feature transform is extracted to increase the contactless verification of Palmprints. As shown in [20], a comprehensive study for explaining different Palmprint recognition methods is presented. Different approaches to feature extraction and algorithms are explored. The dataset that recognizes the Palmprint can be categorized as constrained, unconstrained, and partially constrained. In constrained acquisition, most Palmprint datasets are verified to be used for feature extraction. Unconstrained acquisition, the images are acquired upon the selection of the user. In partial acquisition, the biometric images are matched to different devices. As shown in [21], a multi-biometric method is applied based on fingerprint and finger-vein to improve and enhance user authentication's accuracy and security. The multi-biometric method is based on combining the fingerprint features with finger-vein features where a feature–level fusion is used to match the performance and security of the proposed method.

Recent research on Palmprint methods uses touchless Palmprint recognition based on weight-based metric learning where only parts of the Palmprint categories are identified during the training process [22].

Another palmprint model based on cross-device recognition is presented in [23]. The authors suggested the cross-device method as an innovative idea for IoT devices that can contain different specifications and characteristics. The research aims to apply a distribution model to decrease the cross-device gap. A new cross-device is developed to measure the efficiency of using a new IoT device to maintain the efficiency of verifying biometric users. Another enhanced biometric recognition for Palmprint is presented [24], where a deep learning algorithm using a convolution neural network (CNN) is applied. The efficiency of recognizing users' Palmprint can be enhanced using balanced loss and contrast loss, and experimental results are conducted on seven different datasets to provide provable palmprint security measures. The authors of [25] proposed a fingerprint and palmprint recognition filtering mechanism for biometric template protection that can protect biometric data from illegal users. A feature called the middle of the triangle is applied by validating the input image quality then the palmprint details or minutiae are extracted from the model to increase the recognition accuracy.

3 Methodology

There are two machine learning techniques for user identification and authentication using biometrics data, such as the face, iris, fingerprint, and Palmprint. The first technique is based on classification: the authentication system is trained to recognize and differentiate users from each other. The drawback of this technique is it needs to be trained every time a new user joins the organization (it works with seen users only). On the other hand, the second technique is based on similarity: the authentication system is trained to take two images and recognize whether these images belong to the same user (similar) or a different

user (not similar). This technique achieves the highest generalization, as it is trained on some users and then can be used to predict mutual exclusive users (work with unseen users). The proposed model in this research adopted the second technique based on user identification and authentication similarity.

The Palmprint is considered sensitive information for every user. If it is stolen, it cannot be revoked or changed. Therefore, securing the Palmprint features stored in authentication systems is vital and challenging. Moreover, securing these templates is mandatory for protecting the raw Palmprint images. This proposed research proposes a model for securing the Palmprint stored templates using Hetero-Associative Memory for Palmprint template translation and projection using matrix multiplication and dot product multiplication. The proposed model is also helpful in securing the user's stored templates when he is enrolled in different authentication systems, as the user joins various organizations which can use the same Palmprint image for identification and authentication. The main objective of this research is to secure the person's Palmprint feature maps (i.e., template) by translating them into an irreversible template that can be stored safely to protect the original person's template from being stolen. The training phase for the proposed model is performed through two stages: the user enrollment stage and the user recognition stage. The steps of the proposed model are discussed in detail in the following subsection.

3.1 User Enrollment Stage

This stage aims to transform the person's Palmprint template into an irreversible template (different feature space) to protect the original person's template. First, the user's anchor image is translated into the features map using the VGG16-Palmprint model, presented in [26]. Then, a Hetero-associative Memory (HAM) is constructed and trained for the user per anchor (during enrollment), as shown in Fig. 1.

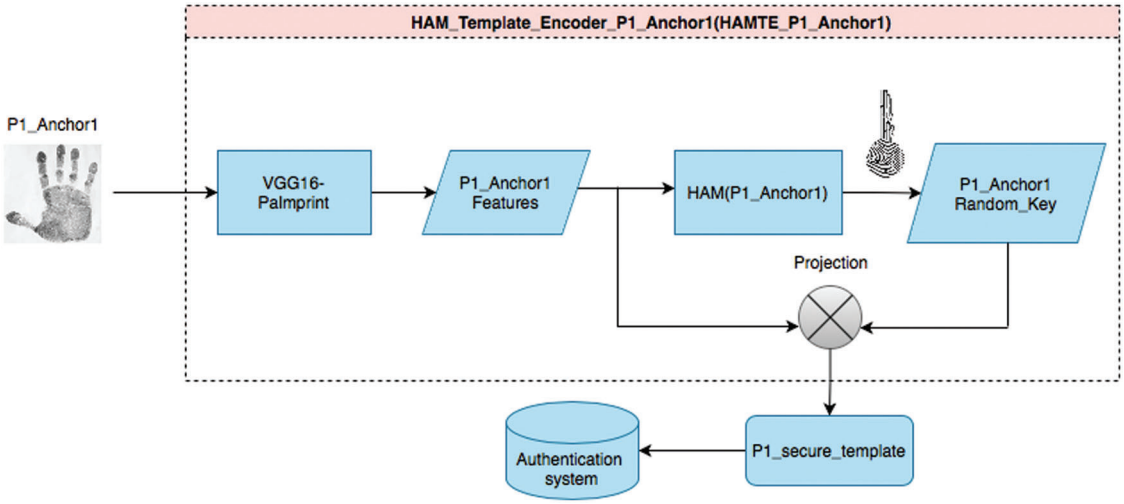

Figure 1: User enrollment stage for HAMTE model

HAM model is constructed and trained for each enrolled user's anchor. A random key is generated for each user's anchor. The generated random key is projected with the user's anchor features map to generate the secured template, which is safely stored in the authentication system database. The projection of the secured key with the original user's anchor features provides the irreversible characteristics for the proposed model, which is explained in detail in the following sections. The network cell presented in this stage is called HAMTE (it means "Hetero-associative Memory for template encoder"). The steps for the enrollment stage of the proposed network cell are presented by the given algorithm, as shown below:

Algorithm 1: HAMTE Network Cell (Enrollment Stage)

Input A set of **m** training Palmprint samples $\{\mathbf{S_1},\ldots,\mathbf{S_m}\}$, h: the size of the encoding key, where h $\in\{128, 256, 512, 1024\}$

1 Generate a random encoding key K_i, for each Palmprint sample/Anchor, S_i

2 Generate feature vector $\mathbf{V_i}$ of size [1 × 4096] using the VGG16-Palmprint model

3 Construct hetero-associative memory of size [4096 $\times h$]

4 Initialize the network's weight matrix, **W**.

5 Update the weight matrix using the given Equation:

$$W = V_i^{-1} \times K_i$$

6 Create E_{ref} by reshaping the vector V_i into a 2-D Matrix of size (o × p), where (o × p = 4096)

7 Create K_{ref} by reshaping vector K_i into a 2-D Matrix of size (p × n), where (p × n = h)

8 Generate the secured template, S_{ref} of size (o × n) matrix using the given Equation: $S_{ref} = E_{ref} \otimes K_{ref}$, [where $\otimes$ indicates the projection operation]

9 Store W, o, p, n, and S_{ref} in the authentication system's database.

It takes the Palmprint images as input and generates the secured template. The Hetero-associative Memory (HAM) is employed in the proposed model to map the input Palmprint features into a randomly generated key. The structure for the HAM is shown in Fig. 2. As shown, the input layer for this network is a matrix of size n representing the input Palmprint feature for a given user. On the other hand, the output layer is a matrix of size m that represents the input randomly generated key for the input pattern. This network is trained to link the input Palmprint features to the associated input key.

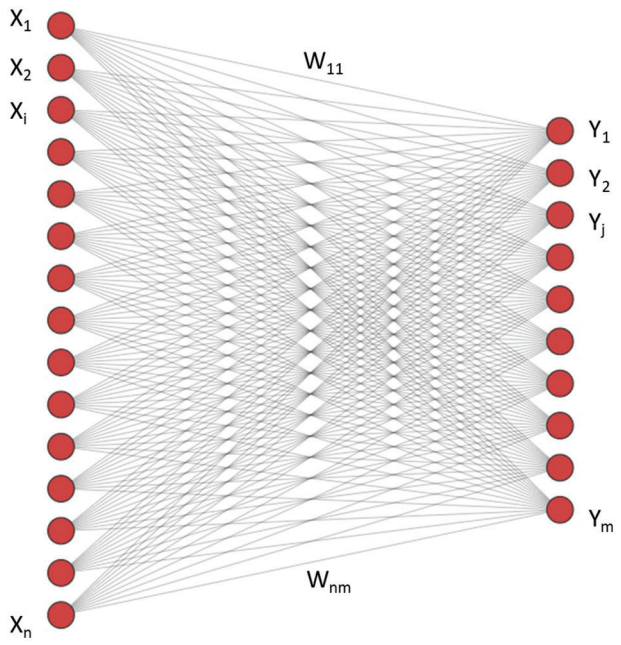

Figure 2: Hetero-associative memory

3.2 User Recognition Stage

This stage aims to train the similarity network to differentiate between the user's Palmprint secured templates of the same or different users. The HAMTE network presented in the first stage is used for generating the secured Palmprint template for either positive (same user) or negative (other users) samples. The data is prepared as pairs of secured Palmprint templates, and the Siamese network is trained to recognize user similarity using these generated secured Palmprint templates. The data preparation and the pairing process are discussed in detail in the experimental result section.

The following example explains the user generation stage in detail. To enroll a subject named: ezz, where A HAMTE model is constructed for every user's anchor (ezz_anchor1, ezz_anchor2, etc.), the secured Palmprint templates are produced, which simulate the enrollment of this user in different authentication systems. Having average eight samples per subject, Each HAMTE model is used for generating the secure templates for 14 pairs; seven positive pairs of subject "ezz", and seven negative pairs of different users: ([Ezz_sec_anchor1, Ezz_sec_sample2], [Ezz_sec_anchor1, Ezz_sec_sample3], [Ezz_sec_anchor1, Foad_sec_sample3], [Ezz_sec_anchor1, Ayman_sec_sample4] … etc.). The generated pairs are used for training and testing the Siamese network for similarity. The Siamese network is trained to correctly classify whether the secure Palmprint sample belongs to the same user or not. The Siamese network is learned to minimize the difference between the positive and the anchor while maximizing the difference between the negative and the anchor. Fig. 3 visualizes the user recognition stage. The steps for the recognition stage of the proposed network cell are presented in the following algorithm:

Algorithm 2: HAMTE Network Cell (Recognition Stage)

Input A testing Palmprint sample S_{test}

Database Retrieval: $\boldsymbol{W_{ref}}$, o, p, n, and $\boldsymbol{S_{ref}}$ from the system's database

1 Generate the feature vector $V_{\textbf{test}}$ of size [1 × 4096] using the VGG16-Palmprint model

2 Generate $\boldsymbol{K_{temp}}$ using $V_{\boldsymbol{test}}$ and the retrieved $\boldsymbol{W_{ref}}$ by applying the following Equation:

$$K_{\text{temp}} = V_{\text{test}} \times W_{ref}$$

3 Create E_{test} by reshaping the vector V_{test} into a 2-D Matrix of size (o × p)

4 Create K_{test} by reshaping the vector K_{temp} into a 2-D Matrix of size (p × n)

5 Generate the Secure test template, S_{test} of size (o × n) matrix using the given Equation: $S_{\text{test}} = E_{\text{test}} \otimes K_{\text{test}}$, [where ⊗ indicates the projection operation]

6 Input both $\boldsymbol{S_{ref}}$ and S_{test} to Siamese similarity Network

7 The Siamese similarity Network responds with either similar or not similar.

Output: Authorized/Non-authorized message

The Siamese network is a neural network that accepts two n-dimensional feature vectors, then passes them to the Euclidean distance layer, which calculates the distance between two vectors. The output from the Euclidean distance layer is zero or around zero in case the vectors are generated from the same user or produce high responses in case the vectors are generated from different users. Two 128 dense layers were added after the Euclidean distance layer, which is learned to produce very few responses when classes are dissimilar while producing high responses when classes are similar. The network uses binary_crossentropy as a loss function and minimizes the difference between similar secured templates while maximizing the difference between different secured templates. This is achieved by maximizing the decision boundary between similar and not similar users.

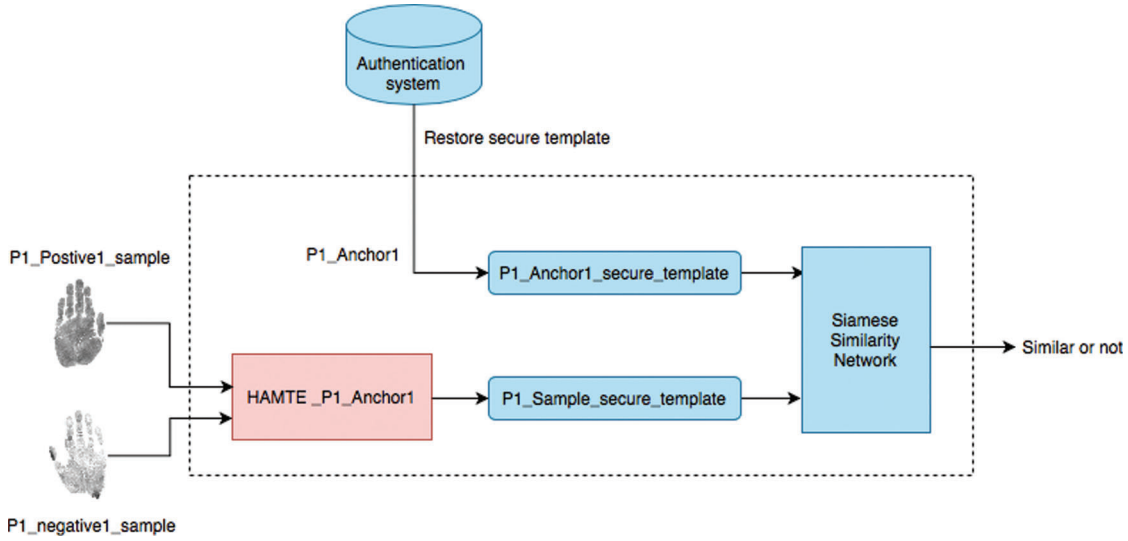

Figure 3: User recognition stage for HAMTE model

4 Experimental Results

In this section, several experiments are conducted to evaluate the proposed Hetero-Associative Memory encoder (HAMTE) performance for Palmprint template protection. The proposed method is tested using the CASIA dataset [27], which contains 306 subjects with eight samples per subject.

4.1 Settings of Experiments

The purpose of the preparations stage is to prepare the Palmprint data as pairs of positive and negative to be used for training the Siamese network for predicting the similarity. The Palmprint data is constructed as pairs in the form (anchor, positive) and (anchor, negative). The dataset is randomly split into training, testing, and validation sets; each data part contains a mutual exclusive subject (users) as follows:

- The training set contains 214 subjects, eight samples per subject, and the total number of samples is 1712.
- The validation set contains 46 subjects, eight samples per subject, and the total number of samples is 368.
- The testing set contains 46 subjects, eight samples per subject. The total number of samples is 368 samples.

The feature extraction stage is performed using the Palmprint pre-trained VGG16 model [26]. Each subject template contains 4096 feature maps extracted from each subject's Palmprint image. Then, the pairs generation stage is conducted by enumerating the subjects' samples as follows: generating positive pairs [anchor, positive]: each sample of the subject (e.g., P1–sample1) is paired with another sample of the same subject (e.g., P1–sample2). Then, generating negative pairs [anchor, negative]: each sample of the subject (e.g., P1–sample1) is paired with another random sample of another subject (e.g., P23–sample3). Every single subject sample is paired with the seven other samples of the same subject and another seven random samples from different subjects. i.e., 14 pairs for every subject/sample gives 112 pairs per subject (56 positive pairs and 56 negative pairs for each subject).

After paring the whole dataset, there are total number of 23,968 pairs for the training, total number of 5,152 pairs for the validation, and total number of 5,152 pairs for the testing. Six experiments are performed

to compare different projection techniques' security complexity and accuracy to achieve the balance between high recognition accuracy and security complexity (irreversible) for each projection technique.

- **Experiment 1** (baseline): In this experiment, the Siamese network is trained using the original Palmprint features map, which is generated from the VGG16. This measure is used as a baseline to compare other experimental results.
- **Experiment 2** (The Random-key size is the same as the template size): In this experiment, the secured Palmprint template is generated using a Random-key with the same size as the original Palmprint features map (4096). The VGG16-Palmprint model generates the original Palmprint features. The Siamese network is trained using the generated secure Palmprint template. The projection technique is pair-wise multiplications by dot product the corresponding vector elements of the Random-key and the original Palmprint feature map. The generated secured Palmprint template is the same size as the original Palmprint features map.
- **Experiment 3** (The Random-key size is 128, with different reshape of the original Palmprint features map, n and m denote the number of rows and columns, respectively): In this experiment, the secured Palmprint template is generated using a Random-key of size equals 128, with different reshape of the original Palmprint features map into different matrices, as shown in Table 1. The projection technique is a matrix multiplication of Random-key with the reshaped matrix of the original Palmprint features map. The size of the generated secured Palmprint template varies depending on the reshaped dimensions of the original Palmprint features map. Then, the Siamese network is trained on the generated secured Palmprint templates.

Table 1: Parameters settings for experiment 3

Generated secured template size	Reshape of original palmprint features (4096)	
	n	m
32	32	128
128	64	64
512	128	32
2048	256	16

- **Experiment 4** (The Random-key size is 256, with different reshape of the original Palmprint features map, n and m denote the number of rows and columns, respectively): In this experiment, the secured Palmprint template is generated using a Random-key of size equals 256, with different reshape of the original Palmprint features map into different matrices, as shown in Table 2. The projection technique is a matrix multiplication of Random-key with the reshaped matrix of the original Palmprint features map. The size of the generated secured Palmprint template varies depending on the reshaped dimensions of the original Palmprint features map. Then, the Siamese network is trained on the generated secured Palmprint templates.
- **Experiment 5** (The Random-key size 512, with different reshape of the original Palmprint features map, n and m denote the number of rows and columns, respectively): In this experiment, the secured Palmprint template is generated using a Random-key of size equals 512, with different reshape of the original Palmprint features map into different matrices, as shown in Table 3. The projection technique is a matrix multiplication of Random-key with the reshaped matrix of the original Palmprint features map. The size of the generated secured Palmprint template varies depending on the reshaped

dimensions of the original Palmprint features map. Then, The Siamese network is trained on the generated secured Palmprint template.

Table 2: Parameters settings for experiment 4

Generated secured template size	Reshape of original palmprint features (4096)	
	n	m
16	16	256
64	32	128
256	64	64
1024	128	32
4096	256	16

Table 3: Parameters settings for experiment 5

Generated secured template size	Reshape of original palmprint features (4096)	
	n	m
32	16	256
128	32	128
512	64	64
2048	128	32

- **Experiment 6** (The Random-key size is 1024, with different reshape of the original Palmprint features map, n and m denote the number of rows and columns, respectively): In this experiment, the secured Palmprint template is generated using a Random-key of size equals 1024, with different reshape of the original Palmprint features map into different matrices, as shown in Table 4. The projection technique is a matrix multiplication of Random-key with the reshaped matrix of the original Palmprint features map. The size of the generated secured Palmprint template varies depending on the reshaped dimensions of the original Palmprint features map. Then, the Siamese network is trained on the generated secured Palmprint templates.

Table 4: Parameters settings for experiment 6

Generated secured template size	Reshape of original palmprint features (4096)	
	n	m
16	8	512
64	16	256
256	32	128
1024	64	64
4096	128	32

4.2 Performance Evaluation

This subsection includes intensive experiments to validate the performance of the proposed Hetero-Associative Memory encoder (HAMTE) for Palmprint template protection under different scenarios. The recognition performance of the proposed model is evaluated in terms of various metrics, such as the FAR (false acceptance rate), FRR (false rejection rate), GAR (Genuine Acceptance Rate), and EER (equal error rate). GAR indicates the number of correct predictions divided by the total number of predictions. FAR denotes when the impostor is marked as an original and is allowed to pass, it can be calculated by Eq. (1). FRR denotes the situation in which the original is rejected, which can be calculated by Eq. (2). GAR denotes the situation in which the original is accepted, and it can be calculated by Eq. (3). EER is the point where FAR and FRR cross (smaller is better); it can be calculated by Eq. (4). The goal is to minimize both FAR and FRR. The accuracy deviation from the baseline is calculated by comparing the test recognition accuracy/EER for each experiment with the test accuracy/EER for the first experiment.

$$FAR = \frac{Imposter\ Score\ Exceeding\ Threshold}{All\ Imposter\ Score} \times 100 \tag{1}$$

$$FRR = \frac{Genuine\ Scores\ Falling\ Below\ Threshold}{All\ Genuine\ Scores} \times 100 \tag{2}$$

$$GAR = 1 - FRR \tag{3}$$

$$EER = \frac{FAR + FRR}{2} \tag{4}$$

Table 5 shows the performance results of the experiments. It is observed that the best-found result deviated from the baseline experiment by around 2.2% and 0.02 for recognition accuracy and EER, respectively. The best-found result is achieved by experiment 2 when the secured Palmprint template is generated using a Random-key with the same size as the original Palmprint features, and the generated secured template size is produced by a dot product projection technique. On the other hand, the worst-found result deviated from the baseline experiment by around 10% and 0.09 for recognition accuracy and EER, respectively. The worst-found result is produced by experiment 4 when the key size and the generated secured template size equals 256 and 16, respectively, and the generated secured template size is produced by a matrix multiplication projection technique.

Table 5: Performance results

Exp. no.	Key size	Secured template size	Validation accuracy	Test accuracy	Validation EER (Equal Error Rate)	Test EER (Equal Error Rate)	Accuracy deviation from baseline	EER deviation from baseline
1	4096	4096	**0.9311**	**0.8915**	**0.0691**	**0.1085**	**-**	**-**
2	4096	4096	**0.9171**	0.8718	0.0827	0.1282	2.2%	0.0200
3	128	32	0.8701	0.8385	0.1300	0.1615	6%	0.0615
		128	0.8906	0.8568	0.1093	0.1432	4%	0.0432
		512	0.8922	0.8605	0.1079	0.1395	3%	0.0395
		2048	**0.8921**	**0.8607**	**0.1077**	**0.1393**	**3%**	**0.0393**

(Continued)

Table 5 (continued)

Exp. no.	Key size	Secured template size	Validation accuracy	Test accuracy	Validation EER (Equal Error Rate)	Test EER (Equal Error Rate)	Accuracy deviation from baseline	EER deviation from baseline
4	256	16	0.8397	0.8065	0.1603	0.1935	10%	0.0935
		64	0.8816	0.8425	0.1184	0.1575	6%	0.0575
		256	0.8940	0.8639	0.1059	0.1361	3%	0.0361
		1024	0.8973	0.8627	0.1027	0.1373	3%	0.0373
		4096	**0.9021**	**0.8633**	**0.0980**	**0.1367**	**3%**	**0.0367**
5	512	32	0.8808	0.8382	0.1192	0.1618	6%	0.0618
		128	0.8853	0.8424	0.1147	0.1576	6%	0.0576
		512	0.9000	0.8565	0.1001	0.1435	4%	0.0435
		2048	**0.9002**	**0.8577**	**0.0998**	**0.1423**	**4%**	**0.0423**
6	1024	16	0.8537	0.8212	0.1462	0.1788	8%	0.0788
		64	0.8863	0.8545	0.1137	0.1455	4%	0.0455
		256	**0.8979**	**0.8631**	**0.1021**	**0.1369**	**3%**	**0.0369**
		1024	0.8933	0.8525	0.1067	0.1475	4%	0.0475
		4096	0.8994	0.8574	0.1008	0.1426	4%	0.0426

Based on the results for experiment 3, it is observed that the best-found result is obtained when the generated secured template size is 2048. The validation accuracy, test accuracy, validation EER, and test EER are 98.2%, 86.1%, 0.1077, and 0.1393, respectively. While for experiment 4, it is observed that the best-found result is obtained when the generated secured template size is 4096. The validation accuracy, test accuracy, validation EER, and test EER are 90.2%, 86.3%, 0.0980, and 0.1367, respectively. For experiment 5, it is observed that the best-found result is obtained when the generated secured template size is 2048. The validation accuracy, test accuracy, validation EER, and test EER are 90%, 85.8%, 0.0998, and 0.1423, respectively. While for experiment 6, it is observed that the best-found result is obtained when the generated secured template size is 256. The validation accuracy, test accuracy, validation EER, and test EER are 89.8%, 86.3%, 0.1021, and 0.1369, respectively.

The results also reflect the impact of the generated template size on the recognition accuracy; it is observed that increasing the size of the generated template increases the recognition accuracy for almost all experiments, which is also helpful from a security perspective. On the other hand, increasing the size of the generated template increases the time complexity for the training and the projection phases.

Fig. 4 shows the recognition accuracy for the validation and test set. As depicted in the figure, the overall best recognition accuracy for the validation dataset was 90.2%, which is achieved for a templet size equal to 4096 and key size equal to 256. In comparison, the worst recognition accuracy for the validation dataset was 84%, which is achieved for templet size equals 16 and key length equals 256. On the other hand, the best recognition accuracy for the test dataset was 86.4%, achieved for templet size and key size equal to 256. In comparison, the worst recognition accuracy for the test dataset was 80.6%, which is achieved for templet size equal to 16, and key size equal to 256.

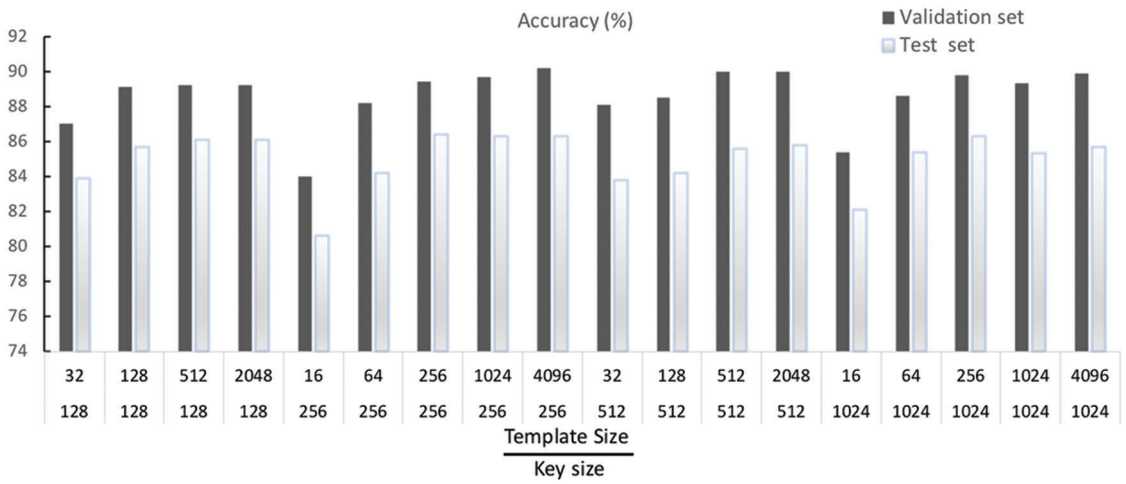

Figure 4: Recognition accuracy for the proposed HAMTE model

Fig. 5 shows the EER for the validation and test set. As shown in the figure, the overall best EER for the validation dataset was 0.09802, it is achieved for templet size equal 4096, and key size equals 256, while, the worst EER for validation dataset was 0.16033, it is achieved for templet size equals 16 and key size equals 256. On the other hand, the best EER for the test dataset was 0.13608, achieved for templet size and key size equal to 256, while the worst ERR for the test dataset was 0.19355, achieved for templet size equal to 16 and key size equals 256.

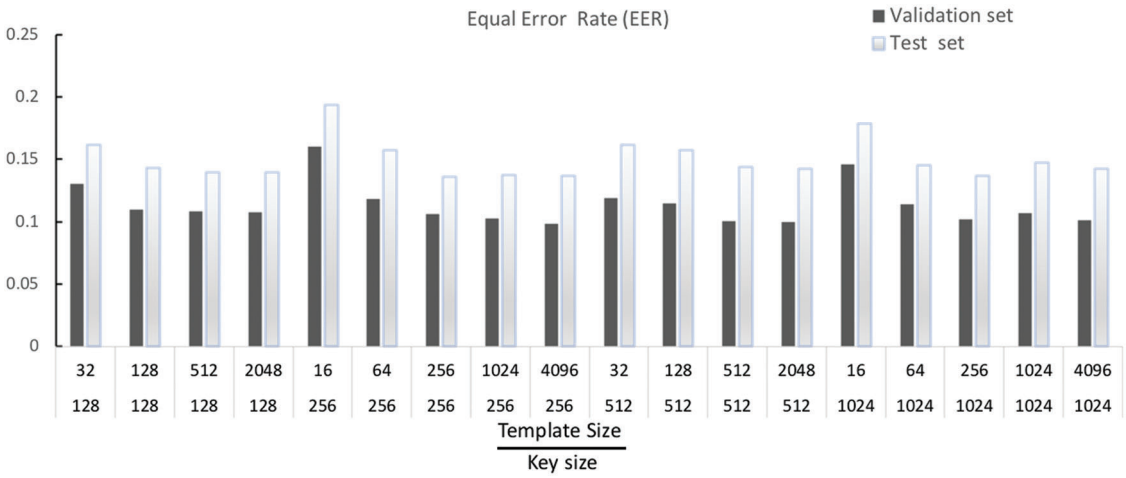

Figure 5: EER for the proposed HAMTE model

A comparative study was conducted to compare the proposed method with other methods in the literature and to situate our contribution. The comparison is illustrated in Table 6. The proposed method achieves reasonable results compared to the existing methods. Although the simulated experiments indicated marginal performance degradation compared to the original/unprotected system (baseline), the applied method produced secure templates for palmprint biometrics that satisfied the diversity and recoverability properties.

Table 6: Comparison results

Method	Biometric type	Test accuracy	Test EER	Accuracy deviation from baseline	EER deviation from baseline
[6]	Face	0.9299	0.0701	1%	0.0076
[7]	Speech	0.9998	1.975×10^{-20}	1%	3.23×10^{-7}
[8]	Speech	0.9958	0.0035	1%	0.0042
[28]	Iris	0.9644	0.0356	2%	0.0178
[29]		0.9102	0.0898	7%	0.0720
[30]		0.9801	0.0199	1%	0.0021
Proposed model	Palmprint	0.8607	0.1393	3%	0.0393

Eventually, this experiment investigates the elapsed run time of the proposed Hetero-Associative Memory encoder (HAMTE) for Palmprint template protection. In order to produce the secured Palmprint template, the input Palmprint feature vector is associated with a randomly generated key using the hero association memory. Then it is projected with its associated random key. The input Palmprint feature vector, A, with size [4096 × 1], and an associated random key is a vector, B, with length [k × 1], where k ∈ {128, 256, 512, and 1024}, the value of k depends on the applied experiment. During the training phase for the HAM model, the weight matrix, W, which is used for the association, is computed using the following Equation:

$$W = A \times B^{-1} \tag{5}$$

The matrix multiplication for $A \times B$ involves multiplying and adding 4096 × K entries in AB. Therefore, the complexity for generating the weight matrix, W, is O (4096 × k). On the other hand, the projection phase requires reshaping for input vectors to generate the secured template, and the input Palmprint feature vector is reshaped to a matrix of size [M × N]. The associated random key vector is reshaped to a matrix of size [N × P], according to applied experiments, as shown in Tables 1–4. The matrix multiplication for both matrixes involves multiplying and adding N terms for each of MP entries in both matrixes. So, the complexity is O (NMP). The total time complexity for generating the secured Palmprint template is O (4096 × k) + O (NMP).

Table 7 shows the run time analysis of producing the secured templates in all experiments. It can be noticed that the fastest experiment needs 13 milliseconds to produce the secured template, and it is obtained when the generated template size equals 32, and the key length is 128. In contrast, the slowest experiment needs 26 milliseconds to produce the secured template, and it is obtained when the generated template size equals 4096 and the key length is 1024. Table 7 shows that increasing template size increases the elapsed run time for generating the secured template for input users.

4.3 Security Analysis Discussion

The security analysis for the proposed Hetero-Associative Memory encoder (HAMTE) for Palmprint template protection is introduced in this subsection. Key security factors such as irreversibility, recoverability, and diversity properties are discussed in detail in the following subsection.

Table 7: Run time analysis

Template size	Key size	n	m	Time (in Milliseconds)
32	128	32	128	**13**
128		64	64	**15**
512		128	32	**16**
2048		256	16	**16**
16	256	16	256	**17**
64		32	128	**18**
256		64	64	**18**
1024		128	32	**19**
4096		256	16	**19**
32	512	16	256	**20**
128		32	128	**21**
512		64	64	**21**
2048		128	32	**21**
16	1024	8	512	**24**
64		16	256	**24**
256		32	128	**25**
1024		64	64	**25**
4096		128	32	**26**

4.3.1 Irreversibility

The analysis conducted in [28] indicates that using hero association memory for mapping the biometric features into the random key is irreversible if the actual output for the memory and the applied key are hidden. Since the actual output for the proposed model is projected using the input Palmprint data into a new space, and the applied key is used only in the enrollment stage without saving it into the system database, there is no way for any attacker to have the actual output or the applied key. Suppose the input Palmprint template for a given user is represented by a matrix (**P**) with dimension [x × y], and the associated random key for that user is represented by another matrix (**K**) with dimension [y × z], the matrices are represented as follows:

$$\mathrm{P} = \begin{bmatrix} p_{1,1} & \cdots & p_{1,y} \\ \vdots & \ddots & \vdots \\ p_{x,1} & \cdots & p_{x,y} \end{bmatrix}$$

$$\mathrm{K} = \begin{bmatrix} k_{1,1} & \cdots & k_{1,z} \\ \vdots & \ddots & \vdots \\ k_{y,1} & \cdots & k_{y,z} \end{bmatrix}$$

The following Equation computes the applied projection in the proposed model:

$$S = P \times K \tag{6}$$

where S is the secured Palmprint template, it is stored in the authentication system database. Hence, S is represented by a matrix with dimension [x × z] as follows:

$$\mathrm{S} = \begin{bmatrix} s_{1,1} & \cdots & s_{1,z} \\ \vdots & \ddots & \vdots \\ s_{x,1} & \cdots & s_{x,z} \end{bmatrix}$$

For each element $s_{i,j} \in S$, it is calculated as follows:

$$\mathrm{s_{i,j}} = \sum_{l=0}^{y} \mathrm{p_{i,l}} \times \mathrm{k_{l,j}} \tag{7}$$

In case of system's database is compromised, the attacker has only the values of matrix S. The attacker may randomly guess the values of the input key and apply the following Equation to get the values of the input Palmprint data (P).

$$P = S^{-1} \times K \tag{8}$$

In the applied experiments, the key sizes are 128, 256, 512, and 1024 real-valued entries. Therefore, the efforts needed by any attacker to randomly guess the real-valued for a key of such size is very complex. For key 128, suppose the real number is only limited to 1000. It needs 1000^128, i.e., e^384 trails, and if the multiplication operation takes ten milliseconds per operation, 1.9e^376 years is required to brute force the Palmprint template. Alternatively, the attacker may randomly guess the values of the input Palmprint data and apply the following Equation to get the input key values (K).

$$K = S^{-1} \times P \tag{9}$$

In all applied experiments, the matrix P contains 4096 real-valued entries. Therefore, the efforts needed by any attacker to randomly guess the value of a matrix of such size is very complex and infeasible. Moreover, the analysis conducted in [18], which is applied for bipolar representation, shows the computational complexity needed to extract useful information using the stored network's weights. It is proven that it is computationally very hard for any attacker to guess the values of the input pattern or the random key using the stored weights. Using real-valued representation for the input Palmprint data and the random key in the proposed Palmprint model maximizes the attacker's efforts for random guessing the input values. The greater the size of the output template size, input Palmprint pattern, and random key, the maximum number of trials to retrieve and extract the original data will increase. As a result, the intruders or hackers will not be able to disclose information from the weight value using a brute force attack. Therefore, the proposed model satisfies the irreversibility property.

4.3.2 Recoverability and Diversity Properties

The recoverability property is satisfied if the proposed model can generate a new secured template for the same Palmprint data if the system is breached. The recoverability of the model is maintained by generating a new random key. This random key is applied on the output layer of the network model. This is considered a flexible method for generating the random key where new connection weights can be generated for the same Palmprint. The new key should be statistically independent to generate a new independent secured template. The new weights are used to generate a different and independent secured template for the same Palmprint data. The proposed Palmprint model allows the use of different keys with different sizes. The key size can be any natural number z where $z \in N$. Therefore, the proposed

scheme supports the diversity property. On the other hand, using different and independent keys to each authentication system allows the same user to enroll his Palmprint data into different authentication systems where his secured stored palm print template will be different and independent among systems. There is no way for any attacker to get any helpful information by correlating the data using multiple authentication systems [18].

5 Conclusion and Future Work

The efficient Palmprint authentication model enables users to enroll their Palmprint data into multiple authentication systems securely. In this study, an innovative HAMTE neural network which is composed of Hetero-Associative Memory for Palmprint template translation and projection, is proposed. The proposed HAMTE model secures the Palmprint data by generating an irreversible template using Hetero-Associative Memory and projection phase. The generated template can be safely stored in the authentication system database. Moreover, the proposed HAMTE model generates different and independent templates using the same Palmprint data, which helps resist cross-matching attacks and enables the user to enroll in different organizations using his invariant Palmprint data. The performance of the proposed HAMTE model has been evaluated against different experiments on the CASIA dataset. The obtained results showed that the proposed HAMTE model achieved a promising result compared to the original unprotected Palmprint data in terms of recognition accuracy, EER, and running time. The recognition accuracy deviated by around 3%, and the equal error rate (EER) by about 0.02 compared to the original data, with appropriate performance (about 13 ms) while preserving the irreversibility property of the secure template. Moreover, the proposed HAMTE model fulfilled the requirements for template protection schemes, including irreversibility, recoverability, and diversity properties. The proposed HAMTE model can be adopted with a multimodal biometric system for future work. Additionally, different neural network structures can be used for generating the secured templates.

Funding Statement: This work was funded by the Deanship of Scientific Research at Jouf University under Grant No. (DSR-2022-RG-0104).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Minaee and Y. Wang, "Palmprint recognition using deep scattering convolutional network," in *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, USA, pp. 1–4, 2017.

[2] N. Sarier, "Multimodal biometric authentication for mobile edge computing," *Information Sciences*, vol. 573, pp. 82–99, Elsevier, 2021.

[3] M. Pasha, M. Umair, A. Mirza, F. Rao, A. Wakeel *et al.,* "Enhanced fingerprinting based indoor positioning using machine learning," *Computers, Materials & Continua (CMC)*, vol. 69, no. 2, pp. 1631–1652, 2021.

[4] A. Anakath, S. Ambika, S. Rajakumar, R. Kannadasan and K. Kumar, "Fingerprint agreement using enhanced Kerberos authentication protocol on m-health," *Computer Systems Science & Engineering (CSSE)*, vol. 43, no. 2, pp. 833–847, 2022.

[5] H. El-Sofany, "A proposed biometric authentication model to improve cloud systems security," *Computer Systems Science & Engineering (CSSE)*, vol. 43, no. 2, pp. 573–589, 2022.

[6] M. Tarek, "Face templates encryption technique based on random projection and deep learning," *Computer Systems Science and Engineering (CSSE)*, vol. 44, no. 3, pp. 2049–2063, 2023.

[7] S. El-Gazar, W. El-Shafai, G. El-Banby, H. Hamed, G. Salama *et al.,* "Cancelable speaker identification system based on optical-like encryption algorithms," *Computer Systems Science and Engineering (CSSE)*, vol. 43, no. 1, pp. 87–102, 2022.

[8] W. El-Shafai, M. Elsayed, M. Rashwan, M. Dessouky, A. El-Fishawy *et al.,* "Optical ciphering scheme for cancellable speaker identification system," *Computer Systems Science and Engineering (CSSE)*, vol. 45, no. 1, pp. 563–578, 2023.

[9] X. Wu, D. Zhang and K. Wang, "Palm line extraction and matching for personal authentication," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. 36, no. 5, pp. 978–987, 2006.

[10] P. Poonia, P. Ajmera and V. Shende, "Palmprint recognition using robust template matching," *Procedia Computer Science*, vol. 167, no. 1, pp. 727–736, Elsevier, 2020.

[11] J. Wan, D. Zhong and H. Shao, "Palmprint recognition system for the mobile device based on circle loss," *Displays*, vol. 73, no. 7, pp. 1–20, Elsevier, 2022.

[12] A. Ignat and I. Pavaloi, "Key point selection algorithm for palmprint recognition with SURF," *Procedia Computer Science*, vol. 192, pp. 270–280, Elsevier, 2021.

[13] L. Fei, B. Zhang, W. Jia, J. Wen and D. Zhang, "Feature extraction for 3D palmprint recognition: A survey," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 3, pp. 645–656, 2017.

[14] B. Sam and S. Sterlin, "A survey on personal identification of left and right palmprint images," in *IEEE Conf. on Information, Communication & Embedded Systems*, India, pp. 1–7, 2017.

[15] D. Zhong, X. Du and K. Zhong, "Decade progress of palmprint recognition: A brief survey," *Elsevier Journal of Neurocomputing*, vol. 328, no. 6368, pp. 16–28, 2019.

[16] S. Zhao and B. Zhang, "Deep discriminative representation for generic palmprint recognition," *Elsevier Journal of Pattern Recognition*, vol. 98, no. 5, pp. 1–11, 2020.

[17] Y. Aberni, L. Boubchir and B. Daachi, "Multispectral palmprint recognition: A state-of-the-art review," in *IEEE Int. Conf. on Telecommunications and Signal Processing (TSP)*, Spain, pp. 1–5, 2017.

[18] L. Fei, G. Lu, W. Jia, S. Teng and D. Zhang, "Feature extraction methods for palmprint recognition: A survey and evaluation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 2, pp. 346–363, 2019.

[19] X. Wu and Q. Zhao, "Deformed palmprint matching based on stable regions," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 4978–4989, 2015.

[20] A. Ungureanu, S. Salahuddin and P. Corcoran, "Toward unconstrained palmprint recognition on consumer devices: A literature review," *IEEE Access*, vol. 8, pp. 86130–86148, 2020.

[21] W. Yang, S. Wang, J. Hu, G. Zheng and C. Valli, "A fingerprint and finger-vein based cancelable multi-biometric system," *Elsevier Journal of Pattern Recognition*, vol. 78, pp. 242–251, 2018.

[22] H. Shaoa and D. Zhong, "Towards open-set touchless palmprint recognition via weight-based meta metric learning," *Elsevier Journal of Pattern Recognition*, vol. 121, pp. 1–12, 2022.

[23] L. Shen, Y. Zhang, K. Zhao, R. Zhang and W. Shen, "Distribution alignment for cross-device palmprint recognition," *Elsevier Journal of Pattern Recognition*, vol. 132, no. 2, pp. 1–25, 2022.

[24] W. Jia, Q. Ren, Y. Zhao, S. Li, H. Min *et al.,* "EEPNet: An efficient and effective convolutional neural network for palmprint recognition," *Elsevier Journal of Pattern Recognition Letters*, vol. 159, pp. 140–149, 2022.

[25] J. Khodadoust, M. Pérez, O. González, R. Monroy and A. Khodadoust, "A secure and robust indexing algorithm for distorted fingerprints and latent palmprints," *Expert Systems with Application*, vol. 206, no. 10, pp. 1–17, Elsevier, 2022.

[26] A. Fawzy, M. Ezz, S. Nouh and G. Tharwat, "Palmprint recognition system using Siamese network and transfer learning," *International Journal of Advanced and Applied Sciences*, vol. 9, no. 3, pp. 90–99, 2022.

[27] D. Zhang, Z. Guo, G. Lu, L. Zhang and W. Zuo, "An online system of multispectral palmprint verification," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 2, pp. 480–490, 2009.

[28] M. Tarek, O. Ouda and T. Hamza, "Robust cancellable biometrics scheme based on neural networks," *IET Biometrics*, vol. 5, no. 3, pp. 220–228, 2016.

[29] C. Rathgeb, F. Breitinger and C. Busch, "Alignment-free cancelable iris biometric templates based on adaptive bloom filters," in *IEEE Int. Conf. on Biometrics (ICB)*, Spain, pp. 1–8, 2013.

[30] D. Sadhya and B. Raman, "Generation of cancelable iris templates via randomized bit sampling," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 11, pp. 2972–2986, 2019.