Tech Science Press

Improved Siamese Palmprint Authentication Using Pre-Trained VGG16-Palmprint and Element-Wise Absolute Difference

Mohamed Ezz, Waad Alanazi, Ayman Mohamed Mostafa*, Eslam Hamouda, Murtada K. Elbashir and Meshrif Alruily

College of Computers and Information Sciences, Jouf University, Sakaka, 72314, Saudi Arabia
*Corresponding Author: Ayman Mohamed Mostafa. Email: amhassane@ju.edu.sa

Abstract: Palmprint identification has been conducted over the last two decades in many biometric systems. High-dimensional data with many uncorrelated and duplicated features remains difficult due to several computational complexity issues. This paper presents an interactive authentication approach based on deep learning and feature selection that supports Palmprint authentication. The proposed model has two stages of learning; the first stage is to transfer pre-trained VGG-16 of ImageNet to specific features based on the extraction model. The second stage involves the VGG-16 Palmprint feature extraction in the Siamese network to learn Palmprint similarity. The proposed model achieves robust and reliable end-to-end Palmprint authentication by extracting the convolutional features using VGG-16 Palmprint and the similarity of two input Palmprint using the Siamese network. The second stage uses the CASIA dataset to train and test the Siamese network. The suggested model outperforms comparable studies based on the deep learning approach achieving accuracy and EER of 91.8% and 0.082%, respectively, on the CASIA left-hand images and accuracy and EER of 91.7% and 0.084, respectively, on the CASIA right-hand images.

Keywords: Palmprint authentication; transfer learning; feature extraction; classification; VGG-16 and Siamese network

1 Introduction

Recently, biometric authentication includes several methods for identifying users based on physical attributes such as fingerprints and facial features [1,2]. On the other hand, authentication systems such as iris can capture images of either one or both human irises with higher resolution to compare them with the existing pattern. Some of these authentication techniques are unacceptable in some situations, particularly in Saudi Arabia [3]. Palmprint is considered one of the advanced methods for authenticating users that verifies whether the users are authentic or not. Different research methods applied authentication mechanisms to improve users' privacy and data integrity on different platforms,

especially in cloud computing. Cloud computing provides a massive infrastructure for storing, executing, and securing users' services [4] with the ability to provide multi-factor authentication methods to maintain user credentials [5]. Palmprint authentication can be applied as one of the essential biometric authentication methods due to its rich, reliable, and exclusionary features [6]. Palmprint textures resist greater damage and rubbish than fingerprint textures [7]. It consists of single points, tiny points, center lines, texture, wrinkles, and ridges, all of which can be used to identify a person's identity. Also, palmprint is user-friendly and has a large surface area that enables the extraction of discriminative features even with a low-resolution image. Using a palmprint makes it easier to compare one palm with another since it contains information like marking, ridges, texture, creases, and delta points, which also are stable and remain unchanged throughout an individual's life. Palmprint, in nature, is more secure and private than other biometric authentication, such as face or iris, which are publicly available on social media as most people share their pictures. Therefore, palmprint has emerged as a viable area of use in the real world [8]. The contact-based biometrics have several drawbacks; these drawbacks include contaminated patterns that are caused by the latent remains in the scanner. Also, oily, moist, or dry palms can increase the failure-to-collection rate in addition to hygienic risks. The contact-based biometrics increase the spreading of the virus due to its contact nature and also needs a special scanner to be attached to the authentication system. The contactless palmprint that can be obtained without contacting a device's surface can be used as a reliable authentication technique. There is a growing need for contactless biometric solutions, and that is due to individual preferences due to the shift in their perceptions around safety and convenience [9]. However, contactless palmprint authentication needs improvement to be practical and applicable.

Many studies have been conducted to improve the accuracy of the palmprint recognition system based on machine learning techniques on a set of palmprint datasets. One of these techniques is transfer learning, which has proved to be highly successful in tackling classification, regression, and clustering challenges in data analysis tasks [10]. Transfer learning refers to applying previously acquired knowledge in a new context. Such a technique can support palmprint recognition accuracy and reduce the training time when a limited amount of labeled data exists [11,12]. Many studies involved transfer learning with one or more machine learning algorithms, such as Siamese network, MobileNet, VGG16, Random Forest, and SVM [13,14]. The Siamese network shares the weights between two tower networks with the same parameters and configuration. Both tower networks might have a duplicate of a sample model, VGG19, VGG16.

The choice of the model, which extracts the discriminative features, is very important in palmprint recognition. Deep neural network methods perform well in the computer vision and recognition field and are mostly employed as feature extraction methods for extracting the best features from the data. The pre-trained deep neural network can be used as the starting point in a new task in a technique known as transfer learning, which is commonly used in deep neural networks, especially when there is a limited amount of data. Fine-tuning neural network using transfer learning is much easier and faster than training the neural network that is initialized with random weights from scratch. This paper uses a transfer-learning model based on Siamese and VGG-16 neural networks for palmprint identification.

The paper's organization is based on depicting the related works in Section 2 by applying machine learning and deep learning techniques to enhance the palmprint accuracy for recognizing users. Section 3 explores the proposed methodology based on a modified Siamese network. Section 4 explains the applied machine learning and deep learning models for analyzing and discussing the achieved results. Finally, Section 5 explores the conclusion and future work.

2 Related Work

Different biometric authentication methods are applied to preserve users' privacy by maintaining their authentication access to systems and devices. Biometric methods such as fingerprints and palmprints are considered major verification methods that can prevent any leakage of users' data. A fingerprinting method for agreement is proposed in [15]. This paper processes the data of different users by building an architecture based on cloud computing that can maintain the confidentiality of users' data on the cloud. The fingerprint mechanism is embedded with Kerberos authentication, which can enhance the user authentication process. As presented in [16], feature extraction of users' data is used to predict users' identities by logging on to the system and verifying them with stored users' data from the database.

As presented in [17], a multi-layer connectivity network is applied to fingerprints to verify user identity. In addition, the proposed multi-layer is learned by using CNN to enhance the accuracy of the identification process. In addition to the previous study, the authors of [18] proposed a biometric authentication framework based on face recognition as it applied three main steps. The first step is based on removing image entropy, while the second step reduces the complexity. In the last step applied, the Speed-Up Robust Feature (SURF) algorithm was used to increase the accuracy.

Several studies on palmprint recognition have been undertaken to date, and several valuable perspectives on improving its performance. Palmprint technology is applied to identify and authenticate users based on their unique features automatically. The main mechanism of palmprint identification and recognition is based on three main steps. The first step is based on acquiring and preprocessing palmprint images. A feature extraction is applied until the palmprint images are stored in the database server. The second step is analyzing unknown images by applying preprocessing and feature extraction of the new images. Finally, feature matching is implemented to match the stored images with the new images after feature extraction.

As shown in [19], the characteristics of palmprint features from the CASIA dataset are defined and extracted where the local frequency locating for the palmprint image is calculated. The weighted average on several scales is measured, and the absolute value is defined to predict each point in the palmprint lines. This method can enhance the image quality by reversing the image histogram. Increasing the performance of authenticating users is also based on the quality of extracted palmprint images that can be affected by different distortions such as wrinkles and pore features that can reduce the identification of palmprint features. As shown in [20], palmprint authentication is applied by combining hand geometry-matched points with palmprint matching. The authors segmented dark images and detected different points on the images, such as wrist, fingertip, and finger valley. The applied dataset in this model is small, and the proposed model needs more experiments with a large dataset to validate the performance.

As presented in [21], Speeded-Up Robust Features (SURF) are used for recognizing palmprint images from different datasets, such as CASIA, IITD, and GPDS. The classification of palmprint features is applied using the KNN algorithm. A model called FIKEN for fixed key points number is proposed for identifying a large number of key points on the palmprint images. The authors of [22] tried to increase the accuracy of authentication by using 3D palmprint images that are difficult to be compromised by unauthorized users. In addition, the surface structure of the 3D images is defined with different mechanisms to increase the image details. The identification process in the paper is based on selecting n palmprints from different databases. The samples that achieved the highest matching score with minimum chi-square distance are considered the best.

The lines in the Palmprint are considered the most significant features that can verify users. Palm veins and impressions can be recognized using Convolutional Neural Networks (CNNs). First, authors in [7] proposed a palmprint recognition system using a transfer learning technique. Their system contains four phases, the first phase handling feature extractions by training the VGG16 model on four different datasets [13,14]. The second phase contains similarity calculation by utilizing Siamese networks, and in the third phase, the system applies one-shot learning testing. The last phase in their system covers the authentication method. Their experimental results show that their proposed palmprint recognition system achieved a high average accuracy of 93.2% and 0.19% EER over the four palmprint datasets. In another study, the authors conducted a comprehensive study by implementing and evaluating sixteen pre-trained models [23]. Their experiments cover three types of datasets: constrained, partially constrained, or unconstrained, according to the constraints imposed by the acquisition procedure. They applied the transfer learning technique by utilizing several training parameters and different pre-trained neural network architectures. The ImageNet dataset is used where it is evaluated so that the recognition rate can be obtained.

The study aimed at estimating the importance of general parameters, evaluating the difficulty of datasets, and estimating the transferability of pre-trained neural networks on palmprint datasets. Their study showed that the MobileNet family had the greatest outcomes because it had the lowest overall error rate. According to the findings, it was found that the preprocessing method was architecture-dependent, while the pooling method was dataset-dependent. On another trend, the authors in [8] conducted a study in which a new feature extraction technique is implemented based on Biologically Inspired Transform (BIT) framework. Aimed at eliminating translation problems, illumination, and rotation caused by palm pose variation, BIT was the biologically inspired transform solution by which the visual neurons' response about spatial frequency and orientation edge could be mimicked. The study's findings revealed that the proposed approach could achieve a high genuine acceptance rate at the same level of false accept rate based on the receiver operating characteristics curves. However, the EERs of BIT on corrupted images are higher than EERs on normal images.

One of the recent research methodologies for using the Siamese network is presented in [24], where two CNNs are applied for deep learning on handwritten input data. The transfer learning is used with the pre-trained AlexNet implemented as a feature extractor. In the second step, the pre-trained data is fine-tuned to improve the performance instead of building the Siamese network from scratch. Another recent method for recognizing the palmprint is proposed in [25], where the recognition of the palmprint is applied with the translation of image-to-image. Each image is translated based on different categories using CNN. The authors of [26] deal with palmprint authentication from a different side, where different attacks are applied on different images to view the vulnerabilities that may compromise the palmprint dataset. This can help reconstruct reinforced palmprint images that are difficult to penetrate.

Eventually, in [27], the authors conducted a study in which palmprint images were recognized using a new recognition system that included four main stages. The first stage of the new palmprint identification method was preprocessing, which improved image segmentation, deleted unwanted pixels, and reduced image size. The second stage was segmentation, which extracted the input palmprint image's Region of Interest. The most successful characteristics were recovered utilizing a proposed methodology that used two methodologies, including local binary patterns and directional features. The classification was the final stage of the palmprint identification system, and it involved a matching procedure between the train and test palmprint images.

Most research methodologies focus on the performance and accuracy of palmprint authentications, considered one of the major biometric authentication methods. Different security and recognition measures are deployed on the palmprint images to safely store and retrieve the palmprint for authenticating users. As presented in [28], a secured mechanism for indexing palmprint and fingerprints with many distortion points. The authors aimed to prevent unauthorized users from accessing the palmprint by converting the unprotected template to a protected one. An enhanced feature called Middle of Triangle Side (MTS) is applied to measure the quality of the palmprint image, and then the input images' details are extracted. The median side of the triangle is calculated to extract the MTS for increasing image recognition accuracy. Another recent methodology for enhancing the recognition of palmprint images is presented in [29], where the image is manually labeled with different key points. These key points are considered the most significant points on the palmprint that can distinguish and verify between different users. A Deep Distillation Hashing (DDH) algorithm is provided for deep recognition of palmprint images based on the selected key points. The enhancement of palmprint recognition is also based on a knowledge distillation that can reduce the complexity of the proposed model. Table 1 summarizes all related studies to this work, explaining the approach, the database used, the implemented technique, the used performance metric, and the achievement of each.

Table 1: A summary of the related studies

Ref.	Approach	Model	Dataset	Achievement	Per. Metric
[7]	• The proposed model applied transfer learning using a pre-trained VGG16 model as a feature extraction part. • The generated features feed the Euclidean distance and sigmoid layers to get the final Siamese similarity measures.	VGG16	CASIA Right CASIA Left	**90%** **88.3%**	Accuracy
[23]	• Pre-trained neural network on the ImageNet dataset (DenseNet family, MobileNet family, Inception family, and VGG family). • Configuration parameters are applied, such as pooling parameters and distance (Euclidean and cosine).	MobileNet	CASIA GPDS	**5.65**	EER

Table 1: Continued

Ref.	Approach	Model	Dataset	Achievement	Per. Metric
[8]	• All palmprint images were preprocessed. • Eliminating translation problems, illumination, and rotation using a new feature extraction technique. • Matching using the classification methods: SVM, CRC, Chi-square distance, and KNN.	BIT	CASIA	**6.23**	EER
[19]	• The palmprint image is defined, and the quality of the image is enhanced using log Gabor analysis. • The accuracy is high for only one-hand side palmprint image.	PCA	CASIA	**97.22**	Accuracy
[20]	• A palmprint authentication is applied by detecting key points in the palm images. • A template-based matching is used for filtering matched points.	MobileNet	CASIA	**5.45**	EER
[21]	• Recognizing palmprint images using the SURG algorithm. • Applying the KNN algorithm for classifying image features from three datasets.	FIKEN	IITD CASIA GPDS	**85.11** **86.16** **85.53**	Accuracy
[22]	• Increasing the accuracy of authentication by using 3D palmprint images that are difficult to be compromised by unauthorized users.	ECST_M	PolyU 3D	**3.37**	EER
[29]	• Enhancing the recognition of palmprint images by using a deep distillation algorithm for key points.	DDH	CASIA	**3.11**	EER

As shown in Table 1, most of the research applied to the CASIA dataset did not achieve more than 90% accuracy on either the left or right palm except [19], which achieved 97% accuracy. This research deals with the palmprint as a normal image classification problem by splitting all person's images among train and test splits. However, this split is suitable for classification problems only where the model is trained to recognize only the persons included in the training. However, for similarity, the problem is different as the model should recognize the unseen persons who are not included in the

training (use exclusive mutual persons on train & test). So from the similarity point of view, this model, which achieved 97%, is considered a cheating model as it was evaluated by the sample from persons included in the training dataset.

In addition, as proposed in our previous work [7], we achieved acceptable measures for all datasets as follows; Tongji achieved around 97%, COEP achieved around 96%, and PolyU-IITD achieved around 93%, while the CASIA achieved around 89% (90% R, 88.3% L) accuracy which is the worst result among other datasets.

Palmprint can be used to uniquely identify individuals as it has wrinkles, ridges, and main lines that are highly discriminative. In addition, the palmprint is user-friendly and has a large surface area that enables the extraction of discriminative features even with a low-resolution image. Using a palmprint makes it easier to compare one palm with another since it contains information like marking, ridges, texture, creases, and delta points, which are stable and remain unchanged throughout an individual's life. Palmprint, in nature, is more secure and private than other biometric authentication such as face or iris, which are publics available on social media as most people share their pictures.

This paper investigates a method for achieving end-to-end palmprint identification by employing the Siamese and VGG-16 [30] networks to extract the convolutional features of two input palmprint images and the top network to derive the similarity of two input palms directly. The VGG16 model is retrained based on the left-hand images of the Tongji dataset, while the Siamese network is trained and tested using the CASIA dataset. The suggested model outperforms comparable studies based on the DL techniques.

3 Research Methodology

To determine whether two palmprint images are similar or not, the pre-trained VGG16 model is used in the Siamese network. The methodology is divided into two stages to distinguish if a set of pair palmprint images are similar or not. Firstly, the pre-trained VGG16-ImageNet (trained on the ImageNet dataset) was modified by replacing the 1000 classification layer with the Tongji persons as a class layer. The retrained on the left-hand images of the Tongji (sessions 1 and 2) until achieving the best accuracy on validation data. The updated VGG16 model (VGG16-Palmprint), which is trained to classify different person's palmprints, is used in the second stage after removing the last classification layer and keeping the second last feature layer as the feature map of any palmprint. Then the VGG16-Palmprint model is involved as an input to the Siamese network by applying the transfer learning principle. The final Siamese model is trained based on the left- and right-hand images of the CASIA, as depicted in Fig. 1. Generally, both stages of the methodology need a few steps that should be fulfilled: dataset preparation, pre-processing, feature extraction, making pair of images, training, testing, and similarity.
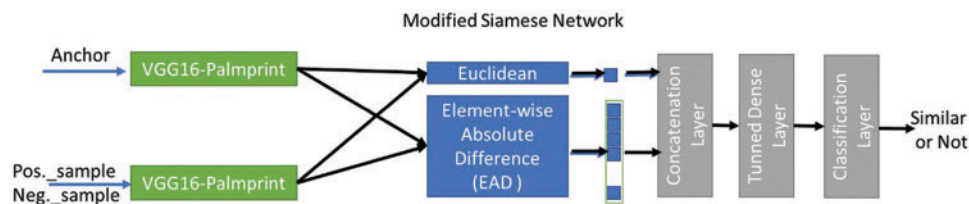

Figure 1: Proposed methodology architecture

The following describe the mathematical of proposed model as shown below:

1. anchorF = VGG16 (Anchor)
2. sampleF = VGG16 (Sample)
3. ec = Euclidean (anchorF, sampleF)
4. ed = EAD (anchorF, sampleF)
5. bothD = ec| ed
6. sim = FullyConnected(bothD)
7. sim? Authenticated: Not

where:

- Anchor: is the stored palmprint template.
- Sample: is the captured palmprint sample to be compare with stored template.
- anchorF: is the feature vector of the stored palmprint template.
- sampleF: is the feature vector of the captured palmprint sample.
- VGG16 function: will take the palmprint and return the extracted features.
- Euclidean function: will calculate the Euclidean distance between anchor and sample features vectors.
- EAD function : will calculate the EAD distance between anchor and sample features vectors
- | : will concatenate two vectors together (distance vectors).
- FullyConnected function: will take the concatenated distance and return the similarity measure.
- sim: is the authentication results system will authenticate the user of the similarity measure; otherwise, authentication will fail.

3.1 Data Preparation

The main objective of this project demands using two standard datasets. There are several fingerprint datasets, including Tongji, COEB, PolyU, CASIA, and IITD [9,31]. In our previous research [7], we discovered that CASIA gives the worst result for recognition, so we decided to give more attention to improving CASIA recognition accuracy and EER [31]. This paper uses Tongji left-hand images to retrain the VGG16 model and create the new transfer learning VGG16-Palmprint model. The left and right images of the CASIA dataset are utilized as input to train the Siamese network model. Tongji dataset is collected from 300 volunteers (192 males and 108 females) to capture both left and right palms in two sessions [9]. The total number of images is 12000, where each person in the session has ten pictures for the left palm and ten for the right. On the other hand, the CASIA dataset holds 5502 palmprint images for both left and right hands and stores them in jpeg format with an 8-bit gray level [32]. Any user in CASIA or Tongji dataset has private folders containing their images, as well as these folders were named with ordered numbers (0001, 0002, 0003, etc.) or mixed numbers and letters (1_left, 2_left, 3_left, etc.). So, the name of each folder is used to label all images inside.

3.2 Data Preprocessing

The process of transforming raw data into a format that can be read, accessed, and analyzed is known as data preprocessing. Before using deep learning algorithms, the preprocessing stage is critical for ensuring or improving a system's overall performance or accuracy. In this model, the collected images need special preparation to be ready for the training and testing stage. At first, the trained VGG16 model on the ImageNet dataset that consists of millions of images required that all input images have a resolution of 224×224. Therefore, the collected images should be resized to the

resolution of 224×224 by using the OpenCV python library [33]. Therefore, we do not involve any extra preprocessing steps to these images, except applying the resizing procedure, making them suitable for the training and testing stage.

3.3 Feature Extraction

The feature extraction stage aims to generate and distinguish features and reduce the number of features in a dataset by generating new feature maps from existing images. Deep neural networks such as CNN are one of the most used techniques to extract features and minimize data dimensionality. The VGG-16 used in this research is based on CNN, and CNN is innovated to minimize image feature engineering. In addition, we have applied deep hierarchical learning, which models abstract representations. The first layers will recognize the low-level features. In contrast, the middle layers will recognize the medium-level shapes, such as the palmprint's texture, wrinkles, main lines, and ridges, while the last layers will be used as an encoder to extract the palmprint features.

Therefore, the pre-trained VGG16-ImageNet model is modified to classify Tongji classes instead of ImageNet classes by retraining the model on a Tongji dataset. The VGG16 model is converted from the general Transfer learning model to specific Palmprint Transfer Learning by removing ImageNet classification (1000 class) and adding the following layers for the modified VGG16-Palm: global average pooling layer, then Dense of 1024, batch normalization, dropout, and finally classification for 300 persons. As shown in Fig. 2, the model is used as a feature extractor by removing the classification layer to be used as a feature extractor in the Siamese network.
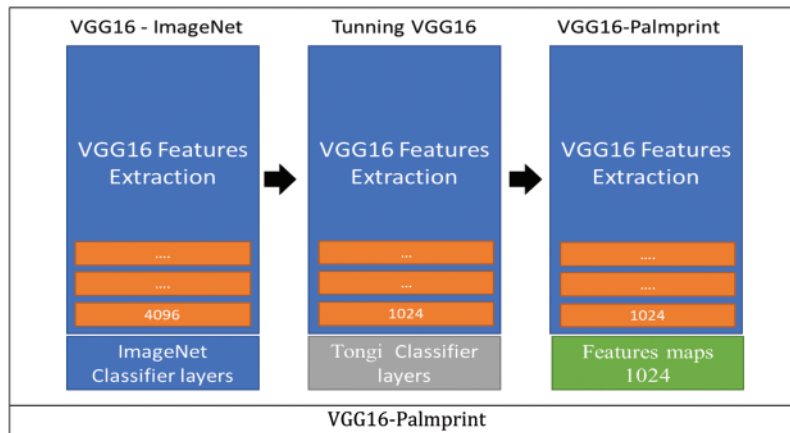

Figure 2: Modified VGG16 palmprint model

3.4 Pairs Generations

The Siamese network accepts two pairs of input data either anchor (i.e., a person palmprint sample) with positive (i.e., the same palmprint sample) or anchor with negative (i.e., a different person palmprint sample).

The procedure of data preparation for making pairs is explained in Algorithm 1. This algorithm uses letter 'N' before an image name ('N_Img3') to denote negative image, while 'P' refers to positive image.

Algorithm 1: Create Pair Procedure

    Input: List_Images [Img_1, Img_2, Img_3, . . . , Img_n], List_Labels1 [lbl_1, lbl_2, lbl_3, . . . , lbl_n].
    Output: List_pair [Img_1, P_Img_2], [Img_1, N_Img_3], [Img_2, P_Img_3], [Img_2, N_Img_3], . . . ,
                 List_Labels [0, 1, 0, 1, 0, 1, 0, . . .] where 0 for same and 1 for different persons.
    Begin
      1. For Index = 0 to Length (List_Images)
          1.1. Find all Images that have the same label as the image in Index = 0, and store them in Same_list = [Img_r, Img_v, Img_x, . . . , Img_k].
          1.2. Find all Images that does not have the same label as the image in Index = 0, and store them in Diff_list = [Img_p, Img_o, Img_n, . . . , Img_z].
          1.3. For Ind = 0 to Length (Same_list)
              1.3.1. IF Ind_P <> Index then
                    1.3.1.1. Add pair [List_Images [Index], Same_list[Ind_P]] to List_pair
                    1.3.1.2. add [0] to List_Labels
              1.3.2. Generate Random index Ind_N
              1.3.3. Add [List_Images[Index], Diff_list[Ind_N]] to List_pair
              1.3.4. Add [1] to List_Labels.
    End

3.5 Training and Testing

The proposed model is divided into two stages: retraining the VGG16 and training the Siamese network. The pre-trained VGG16 model is trained by applying fine-tuning to make one or more layers trainable. A set of layers are adjusted to be trainable starting from layer 19 to 15, as shown in Table 2. This adjustment is applied only to layers located before the fully-connected layers of the classification layer.

Table 2: Fine-tuning the VGG16 layers before the fully-connected layer

No.	Fine-tune Parameter	Trainable Layer
1	19	All layer before the full-connected layer is made untrainable
3	17	'block5_conv3' and 'block5_pool'
4	16	'block5_conv3', 'block5_pool', and 'block5_conv2'
5	15	'block5_conv3', 'block5_pool', 'block5_conv2', and 'block5_conv1'

Additionally, a VGG-16 model is trained on the ImageNet dataset to categorize 1000 classes. Therefore, the top layer of the trained model is removed, and then a new top is added to classify the Tongji classes. Then, the VGG16 is fine-tuned, as demonstrated in Table 2, to get the best VGG16-Palmprint model. The Tongji left dataset for the VGG16-Palmprint is divided into 90% as training data and 10% as test data used in early stopping during training. Then the top layer (classification) of VGG16-Palmprint is removed and used as a feature extractor in the Siamese model. The final Siamese model consists of two VGG16-Palmprint models with the same weights and configuration, as depicted in Fig. 2. Two different distance measurements are applied in the Siamese model; EAD (Element-wise Absolute Difference) and Euclidean distance.

Cosine similarity and Euclidean distance are used interchangeably. However, cosine similarity is better when the magnitude of features does not matter, while, Euclidean distance is when the magnitude is important. In the proposed Siamese similarity network, the magnitude of the distance was important in addition to the individual element-wise distance (EAD), which added more distance parameters as input to the classification layer of the network. This gives more learning parameters by having many outputs from EAD in addition to Euclidean parameters, which are fed to the classification layers (benefits from the no-linear of NN) to improve the classification accuracy.

Euclidean distance is added for calculating the similarity of images by subtracting the difference between two output vectors of these images, while EAD gets the absolute value of subtracting element vectors elements as follows in Eqs. (1) and (2):

$$Vector_1 = \{ fe_{11}, fe_{12}, fe_{13}, \ldots, fe_{1n} \}, \ Vector_2 = \{ fe_{21}, fe_{22}, fe_{23}, \ldots, fe_{2n} \} \tag{1}$$

$$EAD = \{ |fe_{11} - fe_{21}|, \ |fe_{12} - fe_{22}|, \ |fe_{13} - fe_{23}|, \ |\ldots|, \ | fe_{1n} - fe_{2n}| \} \tag{2}$$

where:

fe_{11}, fe_{12}, fe_{13}, etc. are the features elements of the anchor.

fe_{21}, fe_{22}, fe_{23}, etc. are the features elements of the sample.

As $vector_1$ is the anchor feature vector, $vector_2$ is the sample feature vector.

$|fe_{11} - fe_{21}|$ is the absolute difference between pairs of corresponding features.

The distance functions provide a way to measure how close two objects are, where objects are represented as vectors. So the distance functions are often used as difference/error between vectors. The Euclidean distance gives one distance representing the magnitudes difference between two objects which is only one value. On the other hand, the Element-wise Absolute Difference (EAD) is similar to the Manhattan distance, except it keeps the absolute individual difference between every corresponding element of objects' vectors. The EAD (as described in Eqs. (1) and (2) is interesting as the number of dimensions is large. To emphasize the absolute individual distance, this distance will try to reduce all errors equally since the gradient has a constant magnitude.

Moreover, a dense layer is added, starting from 128 to 1024 neurons for searching the best parameters. The 'relu' is used as an activation function, and 'l2' as a kernel regularizing. The regularizer' l2' is a technique that reduces the overfitting problem by penalizing the large weight values. Finally, the output layer uses one neuron and the sigmoid activation function, which scores 0 for matched pair (anchor, positive) of images and 1 for the unmatched pair (anchor, negative). The Siamese model is ready for the training and testing stage, but the input images must be matched or unmatched pairs. The Siamese network dataset uses the first 210 persons' images for training. The next images for persons starting from 211 to 255 are used for validation, and the images of persons from 256 to 300 are utilized for testing. Both models are trained with 200 epochs, and early stop and storing of the best model are also applied.

3.6 Similarity Measure

The similarity principle refers to the ability to find similarities between two vectors. In this work, Euclidean distance and Element-wise Absolute Difference are used. Furthermore, the classification report is used to assess the effectiveness of the proposed model, which consists of recall or sensitivity, accuracy, precision, and F1-score metrics, and they are computed as follows:

Sensitivity: is given by the following equation:

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Accuracy: is provided by the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Precision: is calculated by the following equation:

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

The F1-score is an average to combine precision and recall into a single metric, and the following equation demonstrates how to compute it:

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

where; TP (True-Positive) means that matched cases are classified as matched correctly. At the same time, FN (False-Negative) depicts unmatched cases incorrectly classed as matched. Furthermore, TN (True-Negative) means that unmatched cases are identified as unmatched accurately. Finally, FP (False-Positive) refers to matched cases incorrectly classed as unmatched.

The Equal Error Rate (EER) is applied, which is the point on a ROC curve where the False Positive Rate (FPR) equals the False Negative Rate (FNR). Generally, the smaller the equal error rate number, the greater the biometric system's accuracy. The ERR is calculated be the following equation:

$$fnr = 1 - fpr \tag{7}$$

$$abs\_diffs = |fpr - fnr| \tag{8}$$

$$min\_index = \min(abs\_diffs) \tag{9}$$

$$ERR = \frac{fpr\ [min\_index] + fnr\ [\ min\_index]}{2} \tag{10}$$

4 Results and Discussion

This study carried out several tests to test the proposed model's detection accuracy. All the retrained experiments of the VGG16 model were applied to the left-hand images of the Tongji dataset. Therefore, two phases of experiments have been conducted in this work. The first phase is based on converting general TL to specific Palmprint TL using the left palmprint images of the Tongi dataset. The second phase advocated using the trained VGG16-Palmprint model as part of the Siamese network for learning similarity instead of classification on the CASIA dataset.

4.1 Phase 1: from General TL to Specific Palmprint TL Using Left Palmprint of Tongi

In this phase, the VGG-16 Palmprint was tuned using Tongi left data using four configurations. These four tuning configurations are applied to generate the best-generated features using the modified VGG16-Palmprint, as tabulated in Table 3.

In the Fine-tuning process, we initialize our proposed model with the weights of the VGG16 model trained in the ImageNet dataset. Using Fine-tuning will enable us to speed up the training process and overcome the small-size dataset problem. We started by retraining the last 4 layers using our dataset

and freezing the other layers. We repeat this process for the last 3 layers and the last 2 layers shown in Table 3. Each time we calculate the performance based on the validation accuracy to determine the best setup that determines which layer should be retrained and which should be feezed.

Table 3: Training and validation accuracy for the modified VGG16 palm model

#	Tunning	Train	Validation
15	block5_conv1 block5_conv2 block5_conv3 Feature_Dense_layer	86.31%	78.33%
16	block5_conv2 block5_conv3 Feature_Dense_layer	86.91%	84.67%
17	block5_conv3 Feature_Dense_layer	97.52%	96.33%
19	**Feature_Dense_layer**	**98.69%**	**98.50%**

The best accuracy of the VGG16-Palmprint model is achieved when freezing all VGG-16 layers and training only the last dense layer, where the highest accuracy for training was 98.7% and 98.5% for validation. Finally, the curve of accuracy/loss for the modified VGG16-Palmprint model when freezing all layers is depicted in Fig. 3.
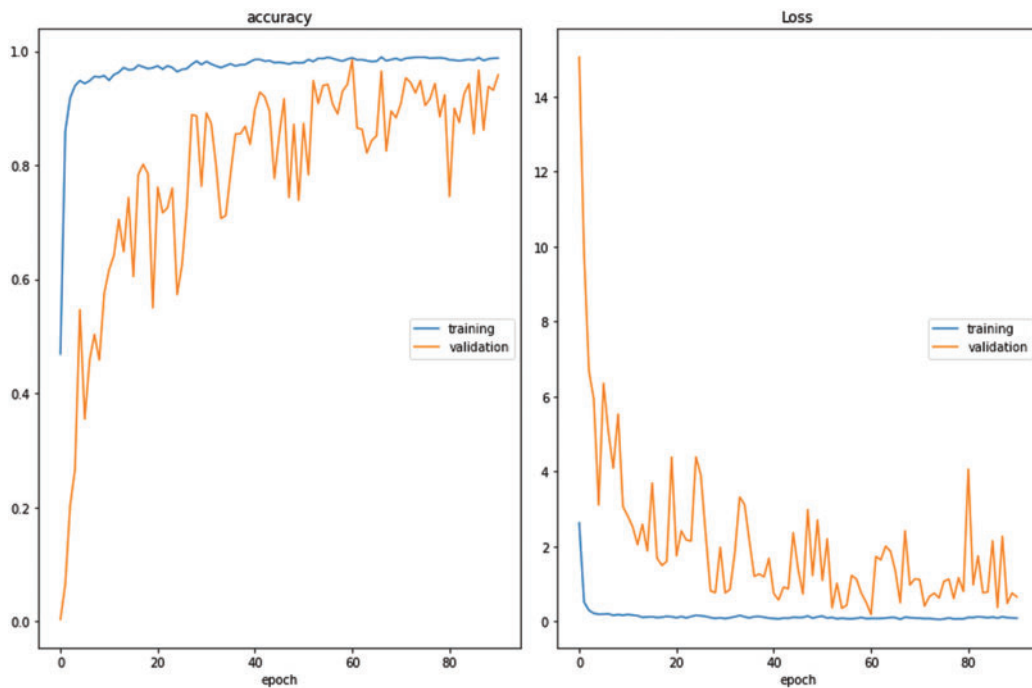

Figure 3: Curve of accuracy/loss of the VGG16 model when freezing all layers

4.2 Phase 2: Using Specific VGG16 Palmprint Model for Learning Similarity

The VGG-16 Palmprint-specific model in the previous phase is built to generate 1024 Palmprint features. In this stage, the model is modified to be part of the Siamese network for learning similarity instead of classification. This is done by using the VGG16-Palmprint to convert the Palmprint image to a features space, then add it as two towers of the Siamese network by freezing these tower weights. It is used as a feature extraction layer. Furthermore, while preparing the data, instead of classifying similar as 1 and dissimilar as 0 and using constructive loss, we classify similar as 0 and dissimilar as 1, which makes the model learn to output 0 as the minimum distance for similarity and 1 as the maximum distance for dissimilar. Then, the Siamese network model is applied using two different distance measurements: EAD (Element-wise Absolute Difference) and Euclidean distance for one experiment, using EAD only; and Euclidean only. The EAD will produce the same demission (1024 neurons) as extracted features from VGG16-Palmprint.

4.2.1 Left Hand Images of CASIA

In the left hand of the CASIA images, the final achieved accuracies for testing and validation are presented in Table 4. Since the best accuracies are gained by using Euclidean and EAD, both measures are used in the second experiment for checking the best dense layer that can give higher accuracy.

Table 4: Validation and testing accuracy for Siamese network with three distance measures on the left images

Description	Validation	Testing
Euclidean only	84.45%	88.98%
EAD only	90.95%	90.58%
Euclidean + EAD	**90.97%**	**90.82%**

In the classification layers of the Siamese network, four different dense layers from 128 to 1024 were applied to search for the best parameters, as shown in Table 5.

Table 5: Validation and testing accuracy for the Siamese network with different dense layers on the left images

Dense Layer	Validation	Testing	Distance
128	91.22%	91.10%	Euclidean + EAD
256	91.37%	91.58%	Euclidean + EAD
512	**91.55%**	**91.71%**	**Euclidean + EAD**
1024	91.23%	91.04%	Euclidean + EAD

Nevertheless, as shown in Table 6 of the classification report, the average accuracy of this model is 92%. The precision, recall, and f1-score for unequal pair images are 93%, 90%, and 90%, respectively, whereas, for matched pair images, are 91%, 93%, and 92%

Finally, the equal error rate (EER) of the Siamese model achieved 0.082, as depicted in Fig. 4a.

Table 6: Classification report of the Siamese model on left images

Classes	Precision	Recall	F1-score
Similar (0)	93%	90%	92%
Not Similar (1)	91%	93%	92%
Accuracy	-	-	92%

(a) Equal error rate (EER) of the Siamese
model on left images

(b) Equal error rate (EER) of the Siamese
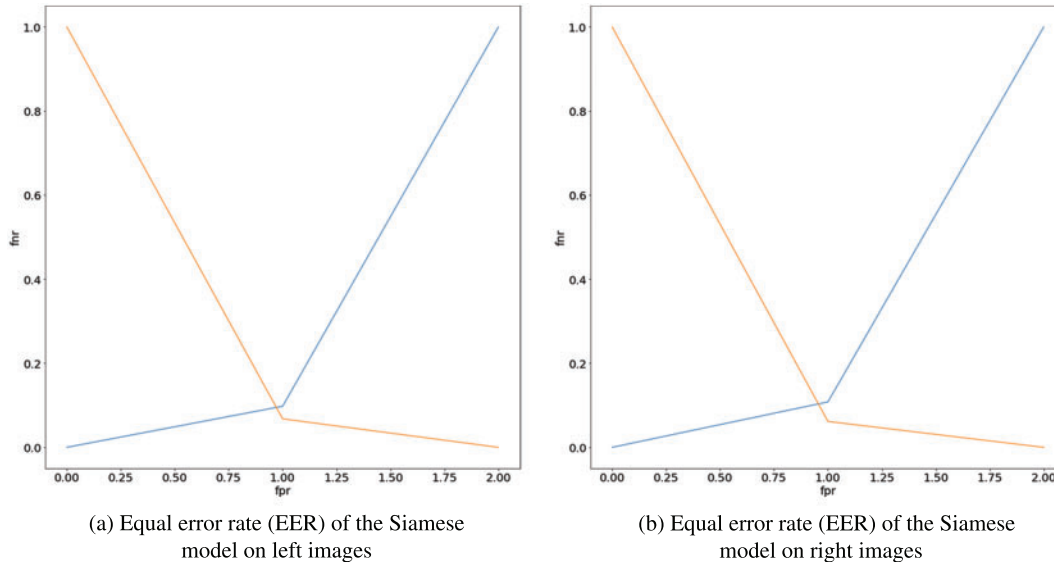model on right images

Figure 4: (a) Equal error rate (EER) of the Siamese model on left images (b) Equal error rate (EER) of the Siamese model on right images

4.2.2 Right Hand Images of CASIA

In the right hand of CASIA images, the final achieved accuracies for testing and validation are presented in Table 7 when applying three distance measures.

Table 7: Validation and testing accuracy for the Siamese network with three distance measures on the right images

Description	Validation	Testing
Euclidean only	84.63%	86.45%
EAD only	88.39%	90.70%
Euclidean + EAD	**88.53**%	**90.68**%

To find the best parameters in the Siamese network's classification layers, four different dense layers ranging from 128 to 1024 were used, as shown in Table 8.

As illustrated in Table 9 of the classification report, this model has an average accuracy of 92%. For unequal pair images, the precision, recall, and F1-score are 94%, 89%, and 91%, respectively, whereas, for matched pair images, they are 90%, 94%, and 92%.

Table 8: Validation and testing accuracy for the Siamese network with different dense layers on the right images

Dense Layer	Validation	Testing	Distance
128	90.26%	91.17%	Euclidean + EAD
256	90.28%	91.25%	Euclidean + EAD
512	**90.41%**	**91.87%**	**Euclidean + EAD**
1024	90.25%	90.84%	Euclidean + EAD

Table 9: Classification report of the Siamese model on right images

Classes	Precision	Recall	F1-score
Similar (0)	94%	89%	91%
Not Similar (1)	90%	94%	92%
Accuracy	-	-	92%

Finally, as shown in Fig. 4b, the Siamese model's equal error rate (EER) was 0.084.

As presented above, several experiments were performed to verify whether or not the final model can recognize a set of paired images as identical pairs on CASIA images, with the aim of improving the similarity prediction accuracy rate. This work aims to enhance the total accuracy achieved by using the VGG-16 and the Siamese network on the CASIA dataset, where the accuracy for the left-hand images was 88.3% and for the right-hand images was 90% [7]. Furthermore, the summary of the outcomes of the proposed model is compared with the related studies, as presented in Table 10.

Table 10: A comparison between the proposed model and other related studies

Study	Dataset	Model	Achievement
[18]	CASIA	MobileNet	EER = 5.65
[7]	CASIA	BIT	EER = 6.23
[5]	CASIA Right CASIA Left	VGG16 + Siamese	Acc = 90 Acc = 88.3
Proposed model	CASIA Right CASIA Left	VGG16 + Siamese	Acc = 91.7, EER = 0.084 Acc = 91.8, EER = 0.082

As presented above, several experiments were performed to verify whether or not the final model can recognize a set of paired images as identical pairs on CASIA images to improve the similarity prediction accuracy rate. This work aims to enhance the total accuracy achieved by using the VGG16 and the Siamese network on the CASIA dataset, where the accuracy for the left-hand images was 88.3% and for the right-hand images was 90% [7]. Furthermore, the summary of the outcomes of the proposed model is compared with the related studies, as presented in Table 10.

The proposed model, as shown in Section 4.1, after applying the fine-tuning VGG16 model, achieved the best accuracy of 98.5% for classifying person palmprint employed as part of the Siamese tower network. Then the fine-tuned VGG16 model was used in the two towers of the Siamese network for feature extraction of both anchor palmprint and sample palmprint.

The two towers of Siamese followed by two distance measures **(Euclidean + EAD)** improved the similarity accuracy of the Siamese network to 90.7% and 90.8% for right and left palmprints, respectively. Then finally, the accuracy of the Siamese network improved to 91.9% and 91.8% for the right and left palmprint, respectively, by adding the 512 dense layers after the distance measure layers, which gives the best-achieved accuracy for the CASIA dataset (left & right) compared to other known proposals as shown in Table 10.

5 Conclusion and Future Work

Palmprint authentication is a robust, reliable, efficient, and promising method of identifying individuals that have rapidly grown in popularity and user acceptance. Palmprint biometrics builds a trustworthy and cost-effective biometric system by using low-resolution images. Several features, such as orientation, texture, fundamental typeface properties, and a fingerprint-like detail feature, make it more precise and dependable. In this work, the Siamese network uses the pre-trained VGG16 model to detect whether or not two palmprint images are similar. The pre-trained VGG16 model is based on the ImageNet dataset and was retrained on the left-hand images of the Tongji dataset until it achieved high validation accuracy. After that, the trained VGG16 model was used as a feature extractor in the Siamese network using the transfer learning principle. Then, the Siamese network was trained on the left and right images of the CASIA dataset. The outcomes reveal that the proposed model outperforms the related studies conducted based on the DL technique by achieving an average accuracy of 92% for both left-hand and right-hand images of the CASIA dataset. Furthermore, ERR of the Siamese model on the left-hand images was around 0.082%, while for the right-hand images was about 0.084%. Consequently, the palmprint authentication system becomes more robust and reliable as the overall accuracy improves. Although the results achieved high accuracy, VGG16 was only implemented as a tower network for the Siamese network. However, other well-known types of CNNs can be used to check the best performance, such as the VGG19, Inception, and DenseNet models. Moreover, it is possible to apply this model to other datasets such as Tongji, COEB, PolyU, CASIA, and IITD for general performance analysis. Future research will try to prepare a palmprint dataset of palmprint in the wild which should capture the palmprint with other objects in the images, such as different backgrounds, objects, and bodies, then check the applicability of our proposal on this new dataset.

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

References

[1] R. Kute, V. Vyas and A. Anuse, "Transfer learning for face recognition using fingerprint biometrics," *Journal of King Saud University–Engineering Sciences*, vol. 33, pp. 1–15, 2021.

[2] 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.

[3]   S. Carlaw, "Impact on biometrics of covid-19," *Elsevier Journal of Biometric Technology Today*, vol. 2020, no. 4, pp. 8–9, 2020.

[4]   A. Alsirhani, M. Ezz and A. Mostafa, "Advanced authentication mechanisms for identity and access management in cloud computing," *Computer Systems Science and Engineering (CSSE)*, vol. 43, no. 3, pp. 967–984, 2022.

[5]   D. Prabakaran and S. Ramachandran, "Multi-factor authentication for secured financial transactions in cloud environment," *Computers, Materials & Continua (CMC)*, vol. 70, no. 1, pp. 1781–1798, 2022.

[6]   L. Shen, B. Liu and J. He, "A boosted cascade of directional local binary patterns for multispectral palmprint recognition," in *Chinese Conf. on Biometric Recognition*, China, pp. 233–240, 2013.

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

[8]   X. Zhou, K. Zhou and L. Shen, "Rotation and translation invariant palmprint recognition with biologically inspired transform," *IEEE Access*, vol. 8, pp. 80097–80119, 2020.

[9]   S. Trabelsi, D. Samai, F. Dornaika, A. Benlamoudi, K. Bensid *et al.,* "Efficient palmprint biometric identification systems using deep learning and feature selection methods," *Springer Journal of Neural Computing and Application*, vol. 34, pp. 12119–12141, 2022.

[10] S. Tammina, "Transfer learning using VGG-16 with deep convolutional neural network for classifying images," *International Journal of Scientific and Research Publications*, vol. 9, no. 10, pp. 143–150, 2019.

[11] H. Xu, L. Leng, Z. Yang, A. Teoh and Z. Jin, "Multi-task Pre-training with soft biometrics for transfer-learning palmprint recognition," *Springer Journal of Neural Processing Letter*, vol. 54, pp. 1–18, 2022.

[12] L. Alzubaidi, J. Zhang, A. Humaidi, A. Al-Dujaili, Y. Duan *et al.,* "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *Springer Journal of Big Data*, vol. 8, pp. 1–74, 2021.

[13] L. Zhang, L. Li, A. Yang, Y. Shen and M. Yang, "Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach," *Elsevier Journal Pattern Recognition*, vol. 69, pp. 199–212, 2017.

[14] X. Liang, Z. Li, D. Fan, J. Li, W. Jia *et al.,* "Touchless palmprint recognition based on 3D gabor template and block feature refinement," *Elsevier Journal of Knowledge-Based Systems*, vol. 249, pp. 1–19, 2022.

[15] 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.

[16] 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.

[17] Y. Chen, J. Liu, Y. Peng, Z. Liu and Z. Yang, "Multilayer functional connectome fingerprints: Individual identification via multimodal convolutional neural network," *Intelligent Automation & Soft Computing (IASC)*, vol. 33, no. 3, pp. 1501–1516, 2022.

[18] R. Srivastava, R. Tomar, A. Sharma, G. Dhiman, N. Chilamkurti *et al.,* "Real-time multimodal biometric authentication of human using face feature analysis," *Computers, Materials & Continua (CMC)*, vol. 69, no. 1, pp. 1–19, 2021.

[19] E. Thamri, K. Aloui and M. Naceur, "New approach to extract palmprint lines," in *IEEE Int. Conf. on Advanced Systems and Electric Technologies (IC_ASET)*, Tunisia, pp. 432–434, 2018.

[20] L. Oldal and A. Kovács, "Hand geometry and palmprint-based authentication using image processing," in *IEEE Int. Symp. on Intelligent Systems and Informatics*, Serbia, pp. 125–130, 2020.

[21] A. Ignat and I. Pavaloi, "Keypoint selection algorithm for palmprint recognition with SURF," in *Int. Conf. on Knowledge-Based and Intelligent Information & Engineering Systems*, Elsevier, Poland, pp. 270–280, 2021.

[22] L. Fei, S. Teng, J. Wu, Y. Xu, J. Wen *et al.,* "Combining enhanced competitive code with compacted ST for 3D palmprint recognition," in *IAPR Asian Conf. on Pattern Recognition*, China, pp. 512–517, 2017.

[23] V. Roşca and A. Ignat, "Quality of pre-trained deep-learning models for palmprint recognition," in *Int. Symp. on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, Romania, pp. 202–209, 2020.

[24] N. Elaraby, S. Barakat and A. Rezk, "A novel Siamese network for few/zero-shot handwritten character recognition tasks," *Computers, Materials & Continua (CMC)*, vol. 74, no. 1, pp. 1837–1854, 2023.

[25] Y. Ma and Z. Guo, "Palmprint translation network for cross-spectral palmprint recognition," *Electronics*, vol. 11, no. 5, pp. 1–10, 2022.

[26] Y. Sun, L. Leng, Z. Jin and B. Kim, "Reinforced palmprint reconstruction attacks in biometric systems," *Sensors*, vol. 22, no. 2, pp. 1–16, 2022.

[27] M. Kadhm, H. Ayad and M. Mohammed, "Palmprint recognition system based on proposed features extraction and (c5. 0) decision tree, k-nearest neighbour (knn) classification approaches," *Journal of Engineering Science and Technology*, vol. 16, no. 1, pp. 816–831, 2021.

[28] J. Khodadoust, M. Medina-Pérez, O. González, R. Monroy and A. Khodadoust, "A secure and robust indexing algorithm for distorted fingerprints and latent palmprints," *Expert Systems with Applications*, Elsevier, vol. 206, pp. 1–22, 2022.

[29] H. Shao, D. Zhong and X. Du, "Deep distillation hashing for unconstrained palmprint recognition," *IEEE Transaction on Instrumentation and Measurement*, vol. 70, pp. 1–13, 2021.

[30] K. Simonyan and A. Zissereman, "Very deep convolutional networks for large-scale image recognition. computer vision and pattern recognition," 2015. [Online]. Available: https://arxiv.org/abs/1409.1556v6.

[31] T. Kumar, E. Julie, Y. Robinson and S. Jaisakthi, "Simulation and analysis of mathematical methods in real-time engineering applications," in *Applied Mathematics in Engineering*, 1st ed., vol. 1, USA: Wiley, pp. 1–386, 2021.

[32] G. Shams, M. Ismail, S. Bassiouny and N. Ghanem, "Face and palmprint recognition using hierarchical multiscale adaptive LBP with directional statistical features," in *Int. Conf. Image Analysis and Recognition*, Portugal, pp. 102–111, 2014.

[33] A. Mordvintsev and K. Abid, "Opencv-python tutorials documentation," 2017. [Online]. Available: https://opencv24-python-tutorials.readthedocs.io/_/downloads/en/stable/pdf/.