Computer Science • 24(2) 2023 https://doi.org/10.7494/csci.2023.24.2.4685

Asma Aounallah
Abdallah Meraoumia
Hakim Bendjenna

LEARNING-FREE DEEP FEATURES FOR MULTISPECTRAL PALM-PRINT CLASSIFICATION

Abstract *The feature-extraction step is a major and crucial step in analyzing and understanding raw data, as it has a considerable impact on system accuracy. Despite the very acceptable results that have been obtained by many handcrafted methods, these can unfortunately have difficulty representing features in the cases of large databases or with strongly correlated samples. In this context, we attempt to examine the discriminability of texture features by proposing a novel, simple, and lightweight method for deep feature extraction to characterize the discriminative power of different textures. We evaluated the performance of our method by using a palm print-based biometric system, and the experimental results (using the CASIA multispectral palm-print database) demonstrate the superiority of the proposed method over the latest handcrafted and deep methods.*

Keywords feature extraction, biometrics, multispectral imaging, deep learning, DCTNet, data fusion

Citation Computer Science 24(2) 2023: 243–271

1. Introduction

Thanks to digitalization, machines can replace manual tasks with computer-vision technologies today in order to solve several problems. In industries, for example, these technologies can automate processes to help reduce costs, eliminate hazardous work, improve working conditions, and perform operations that cannot be controlled manually [24]. For the implementation of such technologies, the data to be processed must be acquired by several sensors that are distributed so as to make it possible to capture the necessary information for controlling a system. Basically, images are the main side of data representation and are often used in robotics, remote sensing, biometrics, and medical imaging. In all of these applications, data-processing algorithms must be embedded in their own machines to access the useful information in the images. In general, these algorithms include pattern-recognition modules that seek to extract discriminating features from images; this has had a considerable impact on the performance of computer vision systems.

A pattern-recognition system (PRS) contains several steps through which its performance can be effectively improved by carefully choosing the method that is used in each of these steps [7, 16]. Acquisition (or capture – typically followed by the preprocessing process), feature extraction, training, and decision are the main steps of the PRS flowchart. In fact, these steps are sequential so that each step utilizes the output of the preceding step as input, often making its precision related to the precision of its predecessor. In general, the data-acquisition and feature-extraction steps are the two most important steps in this type of system, as the training and decision-making steps are efficient if the features of the samples of different classes can be represented separately. In our work (and based on a biometric system), we will therefore try to highlight the impact of multispectral imaging technologies (used in the acquisition step) and the deep-analysis principle (used in the feature-extraction step) on PRS performance; in doing so, we will propose a new deep-feature-extraction method from biometric modality images.

The proposed method is based on the principle of convolutional deep analysis using pre-defined learning-free filters that are determined by the discrete Fourier transform (DFT) technique. This is considered to be a simple and light deep-analysis method – especially in the formulation of filters. Indeed, these filters do not need a training base but are formulated directly using DFT basis functions; this means that the weights of these filters are predetermined and do not need to be modified by the classifier that is used during the training phase. Unlike machine learning, the classifier in deep learning plays the role of an expert in the feature-extraction step that will maximize the recognition rate (deep features). For this (and in order to give this property to our proposed method), the conversion thresholds (used in the binarization layer) are chosen by using optimization algorithms. The effectiveness of the proposed feature-extraction method was examined by using a biometric system, as it is a typical pattern-recognition application.

In the design step during a biometric system's implementation, an important question arises: on which biometric modality will we work? Currently, many biometric technologies have been developed; human hand-extracted features have confirmed their reliability, acceptability by the user, and low cost in all security applications [11, 18] as a consequence of their simplicity and effectiveness in extracting features (even from low-resolution images) [6]. Palm-print modality is one of the human hand-extracted biometric technologies that has received growing attention recently [43]. It is relatively stable, and the image of the hand from which they are extracted can be acquired relatively easily. In fact, some other important traits of the human palm can be used to ameliorate a biometric system's performance. With the multispectral imaging's modern evolution, however, these traits are captured by using infrared-acquisition devices that supply palm-vein images instead of using visible light [23]. These traits feature low risk of falsification, difficult duplication, and stability, as they are under the skin. Moreover, the availability of a device that is able to acquire two types of modalities at the same time has pushed researchers to fuse these two modalities in an attempt to improve unimodal biometric systems by replacing them with multimodal biometric systems [5]. For this, we use the multispectral palm-print images in this work (with several spectral bands) to design a unimodal/multimodal biometric system.

The rest of the paper is structured as follows: Section 2 presents a literature survey on multispectral palm print-recognition systems; Section 3 describes the formulation procedure of convolution kernels through DFT (discrete cosine transform [DCT] and discrete sine transform [DST]) as well as the concept of our proposed deep-feature-extraction method; Section 4 evaluates the proposed method by using a publicly available database (CASIA multispectral database); Section 5 aims to present a comparative study between our proposed feature-extraction scheme and some new methods in the literature; and Section 6 provides the reached conclusions and future scopes.

2. Literature survey

In this part, we will try to highlight the latest work that has been carried out in biometric-recognition systems that use multispectral-imaging technology, and we will address (in particular) the works that have been evaluated by using the CASIA database. Due to the many feature-extraction methods that are used, we will therefore try to divide this overview into two sub-parts: the first concerns hand-crafted methods (feature engineering), while the second presents methods that are based on deep learning (deep feature). Thus, Table 1 summarizes some important works in this area of research. In this table, the performance is given by the genuine acceptance rate (GAR) at the balanced operating point (equal error rate [EER] point).

Ali M.M.M. *et al.* [2] proposed a multimodal palm print-recognition system based on a multi-algorithm scenario. In this work, the feature-extraction method was performed using both local binary pattern (LBP) and two-dimensional locality-preserving projection (2DLPP).

Table 1
Most important biometric systems based on multispectral palm-print modality

Years	Feature extraction	Data set used	GAR [%]
2017 [2]	LBP + 2DLPP (KNN)	CASIA-P	98.55
2018 [4]	CLBP (KNN)	CASIA-MP	100.00
2019 [40]	HDPLS (KNN)	Self-Built-P	99.59
		CASIA-MP	99.97
2020 [21]	LDDBP (KNN)	PolyU-P	99.85
		IITD-P	97.81
		GPDS-P	97.70
		CASIA-P	97.27
2020 [3]	DBM (KNN)	PolyU II-P	100.00
		PolyU MP	100.00
		IITD-P	95.40
		CASIA-P	93.90
2020 [35]	Log-Gabor + KPCA (KNN)	CASIA-MP	99.50
2020 [17]	LDBC (SVM)	PolyU-MP	99.57
		CASIA-MP	95.72
2020 [9]	CBD (SVM)	CASIA-MP	99.75
	TP-LBP (SVM)		98.88
	FP-LBP (SVM)		98.38
2019 [36]	PVSNet	PolyU-P	98.78
		IITI Vein	97.47
		CASIA-MP	85.16
2018 [8]	DCTNet	PolyU-MP	100
		CASIA-MP	99.33
2020 [37]	U-Net	CASIA-MP	99.53
2017 [29]	PCANet	PolyU-MP	100.00
		CASIA-MP	99.88
2019 [33]	DBN	PolyU-MP	99.96
		CASIA-MP	99.67
2019 [22]	PalmNet (Gabor, GaborPCA)	Tongji	99.84
		REST	96.41
		IITD	99.48
		CASIA-P	99.29
2020 [41]	JCLSR	IITD	98.17
		PolyU MP	100.00
		GPDS	96.33
		CASIA-P	98.94
2021 [19]	GPWLD	PolyU-MP	99.99
		CASIA-P	99.76
2021 [15]	Lightweight-CNN	PUT vein	100.00
		CASIA-MP	99.94

The preprocessing process for locating the region of interest was performed using competitive hand valley-detection (CHVD) methods. To achieve the multi-algorithm system, the two feature vectors are combined at the feature level by the concatenation method. When using the CASIA database, the experimental results showed that we achieved the best accuracy result (98.55%) by fusing LBP and 2DLPP. Likewise, the scheme that was proposed in [4] was also evaluated by using a single spectral band from the CASIA database (with a wavelength of 940 nm). In this work (and in order to extract its discriminate feature), the compound local binary pattern (CLBP) was used. In this work and in order to extract its discriminate feature, the Compound Local Binary Pattern (CLBP) was used, it employs both the sign and the inclination information of the differences between the center and the neighboring gray values, for all the P bits that were encoded by LBP, corresponded to a neighbor of the local neighborhood, for that, an extra bit is added in order to build a robust feature descriptor. In the experimental phase, the unimodal system and the multimodal system (a fusion of the right and left palms at the score level) were evaluated.

Wu *et al.* proposed a practical and low-cost palm vein-identification system in [40]. In their scheme, near-infrared (NIR) images of the palm and veins were captured by using a complementary metal-oxide-semiconductor camera instead of an NIR charge-coupled-device camera to dramatically reduce costs. Concerning the region of interest (ROI) determination, the authors adopted the thenar region on the hand/palm (as it is rich in palm veins) to avert the palm-print effect on the captured images and, therefore, on the system's performance. To boost the recognition accuracy, the Haar wavelet decomposition and partial least squares algorithm (HDPLS) were used on the ROI image in order to discriminate the palm-vein feature extraction. In order to assess the effectiveness of the proposed scheme in this work, experiments were carried out on two databases: a self-constructed database, and the public CASIA multispectral database. Unfortunately, the authors did not use all of the spectral bands in the CASIA database (six bands) and only used a single band with a wavelength of 850 nm.

In order to describe the discriminative power of different directions, Fei *et al.* [21] proposed a new exponential and Gaussian fusion model (EGM). For a complete representation of the palm print-direction features, a local discriminant direction binary pattern (LDDBP) was also proposed. Guided by EGM for palm-print representation and recognition, an LDDBP-based descriptor can be formed by exploiting the most discriminant directions. The experimental results explained that the proposed LDDBP method that was conducted on the four widely used palm-print databases (PolyU, GPDS, IITD, and CASIA) showed superiority over some of the others that were tested in this study. In the case of the CASIA database, the recognition rate remained below 97%; this could be improved by combining certain spectral bands.

A fast palm print-recognition method based on the different of block means (DBM) was proposed by Almaghtuf *et al.* in [3]. The method begins with a preprocessing step that extracts a region-of-interest sub-image. After this, the differences

between overlapping block means were computed for a palm print-information description in each direction. In order to obtain a final palm-print code in each direction, a threshold was ultimately applied. In the evaluation phase, a set of experiments were carried out on three publicly available palm-print databases: CASIA, PolyU II (for multispectral palm-print images), and the IIT Delhi Touchless palm-print database. Despite the fact that the proposed scheme used basic operations (*i.e.,* mainly additions and subtractions) in order to reduce the cost of the calculations, the recognition rate unfortunately remained modest (<95%) in the CASIA database.

Thamri *et al.* [35] proposed a multimodal biometric-recognition system that was based on the fusion of the spectral bands of palm-print modalities. In this method, the fusion of information from the same left and right spectral bands was performed at the image level using the 2D discrete wavelet transformation (DWT) technique. After this, Log-Gabor transformation was used as a feature-extraction method, followed by the kernel principal component analysis (KPCA) technique for reducing the feature size. The experimental results that were carried out on the CASIA multispectral palm-print database showed the best recognition rate (99.50%) with white (WHITE) spectral bands.

A novel cross-spectral matching system for identity verification based on RGB (red and blue) and NIR image spectral bands of both palm prints and palm veins was proposed in [17]. First, both palm-print and palm-vein features were extracted from the RGB and NIR palm images in which simplified Gabor filtering (SGF) with a morphological-like operation was used to extract palm-vein lines. Additionally, a simplified LBP encoding scheme called local directional binary code (LDBC) was proposed for both the RGB and NIR palm-print templates. After this, the two feature-extracted sets for each modality were fused at the score level in a way that the proposed system could be flexibly operated as a unimodal/multimodal system. The experimental results that used both the multispectral PolyU-MP and CASIA-MP data sets showed the effectiveness of the proposed system with high accuracy and significant equal error rate (EER) enhancement.

Bouchemha *et al.* [9] proposed a handcrafted feature-learning method for multispectral palm-print representation and recognition based on local distinctive image descriptors (called compact binary descriptors [CBDs]). First, a projection matrix and codebook were used during the training as prior knowledge by using the pixel difference vectors (PDVs) of non-overlapping sub-blocks. Second, the extracted PDVs were encoded into binary codes in the test step that used the projection matrix; then, they were pooled as a histogram feature by using the codebook. Experiments that were carried out on the CASIA multispectral palm-print database showed an improvement of 0.250% with EER when using the WHITE spectral band.

On the other hand, a diverse set of deep-learning models have been developed to learn useful feature representations that have been widely used in face-based biometric systems. Unfortunately, a few types of these models have tried to build a palm print-based biometric system despite the effectiveness of the palmrint modality. Thus, a new

way was proposed in [36] in order to design a palm vein authentication Siamese network (PVSNet) end-to-end deep convolutional neural network (CNN) framework. This model works in two main steps: first, an encoder-decoder network is used for generative domain-specific feature learning; then, pre-trained convolutional layers are applied in an unsupervised fashion as an auto-encoder by a Siamese network. After this, the adjusted triplet loss function for learning-feature embeddings was used to train the proposed model in a way as to minimize the distance between the embedding-pairs from the same subject. On the other hand, a distance maximization was applied with those from different subjects (with a margin). In the experiments, the proposed network was evaluated using the three typical palm-vein databases (CASIA, IITI Vein, and PolyU). The result that were obtained from the CASIA database (a correct recognition rate of 85.16%) indicated the poor performance as compared to the other databases.

A simple and efficient deep-learning-based feature-extraction algorithm called a discrete cosine transformation network (DCTNet) was proposed in [8]. In this study, the effectiveness of the proposed approach (unimodal and multimodal systems) was assessed using two publicly available databases: CASIA, and PolyU. The obtained results clearly indicated that a feature-extraction technique based on DCTNet deep learning can achieve a performance that is comparable to the best advanced techniques.

In [37], Wang *et al.* proposed an end-to-end convolutional neural network to extract the characteristics of palm veins for use in a personal identity-verification system. The training data set was generated by labeling palm pixels by combining some handcraft-based palm-vein image-segmentation methods. The output of the trained U-Net represents the probability that the pixels belong to the vein pattern. To obtain the vein network patterns, a new scheme for encoding the outputs of U-Net was proposed. The experimental results (using a single spectral band) of the public CASIA multispectral palm-print database indicated the effectiveness of the proposed method (which produced an error of 0.47%).

A combination of palm-print multispectral imaging with the PCANet deep-learning feature-extraction method was proposed in [29]. A performance evaluation of the proposed scheme that was carried out on the PolyU and CASIA multispectral databases showed a notable improvement over the system's accuracy.

In [33], Huafeng *et al.* proposed an iterative deep-belief network (DBN) in order to extract vein features to iteratively correct those features, which are generated automatically by using a beforehand very limited knowledge. For the automatic labelization of vein and background pixels, a known handcrafted vein-image-segmentation method was used based on patches that were centered on labeled pixels in order to build the training data set. This data set was used to train the DBN to predict the probability that each pixel belongs to a vein-net. During the iterative procedure, incorrect training data labels are statistically corrected by learning the differences between vein patterns and background ones; the deep-belief network can correctly learn

what a vein pattern is. A notable improvement was shown in the experimental results regarding hand-vein verification accuracy wqhen using two public multispectral palm-print databases (CASIA and PolyU).

To attain high recognition accuracy with touchless palm-print samples that are captured by utilizing several devices without the need to train class labels or utilize pre-trained filters, an effective deep scheme called PalmNet was proposed by Genovese *et al.* in [22]. In this scheme, a new CNN was represented for a palm print-specific filter setting using a recently developed method by an unsupervised procedure without class labels in the training phase based on Gabor responses and principal component analysis (PCA). For applying Gabor filters in a CNN, extract very discriminative palm print-specific descriptors, and adapt to heterogeneous databases, the novel PalmNet method was used. This approach was validated using multiple palm-print databases that were captured using different touchless acquisition procedures and heterogeneous devices; in all cases, significant improvement was demonstrated in terms of recognition accuracy.

In [41], deep convolutional neural networks (DCNNs) were used by following a regular sampling of the various local regions of a palm-print image to learn complete and discriminative convolution features. A joint constrained least-square regression (JCLSR) framework was presented to utilize the commonality of several patches in which a representation for every local region of the same palm-print image was performed; this required having similar projected target matrices for all of the regular local regions of the palm-print image. In palm-print recognition, the under-sampling classification problem can be well-solved by the proposed method, where the proposed JCLSR that was conducted on the IITD, CASIA, noisy IITD, and PolyU multispectral palm-print databases outperformed classical palm print-recognition methods and some subspace learning-based methods for palm-print recognition in the experimental results.

In [19], El-Ghandour *et al.* proposed a new methodology for feature extraction that was a combination of Gabor features with positional Weber's local descriptor (PWLD) features that was known as Gabor-positional Weber's law descriptor (GPWLD). First, a Gabor filter was used with different orientations to capture the salient rotational features that could be found in the output feature maps of a palm-vein image. Second, a uniform division of each feature map into several blocks was made to achieve spatiality. Then, a Weber's law feature descriptor (WLD) histogram was computed for each block in each feature map. Finally, these histograms were concatenated to compose the final feature vector; then, a feature-size reduction was applied using the PCA algorithm to be classified later with a deep neural network (DNN) that included an optimized stacked autoencoder (SAE) with Bayesian optimization and a softmax layer. The experimental results using the CASIA database showed that the proposed method gave the best result (with an error of 0.24%).

A lightweight deep-learning network framework for palm-vein recognition based on a deep convolutional neural network (CNN) was proposed in [15] in which the

applied ROI positioning after palm-image input was able to resist a certain degree of rotation and displacement, reducing the system errors that were caused by the user during the acquisition step. In addition, an adaptive Gabor filter with enhanced imaging features and a triplet loss function that captures sufficient palm-vein data were proposed. The set of experiments were carried out on two multispectral palm-print databases: CASIA (with 100 users), and the PUT database (with 50 users). The results showed that the proposed method required fewer parameters as compared to other methods and had a good recognition error rate of 0.0556%.

3. System framework

In this section, a general framework for constructing a biometric system is proposed that is capable of deeply analyzing a biometric modality by using a series of filters that are derived from DFT. The system's design framework brings together a set of deep-feature-extraction methods that can be used together (using the fusion principle) or separately.

3.1. Outline of proposed approach

This section develops a general overview of feature extraction using the principle of convolutional deep analysis that focuses on the image of a biometric modality as an example. Overall, the components of the proposed method are *(1)* the analysis step, *(2)* the pooling step, and *(3)* the observation vector (feature) step. The analysis step consists of filtering an input image according to several stages by using predefined convolution filters. The role of the second step is to reduce the data size by quantization and encoding processes. The last step is to extract a feature vector from the formed images and convert it to a flat vector (flattening). The outline of the proposed approach is illustrated in Figure 1.

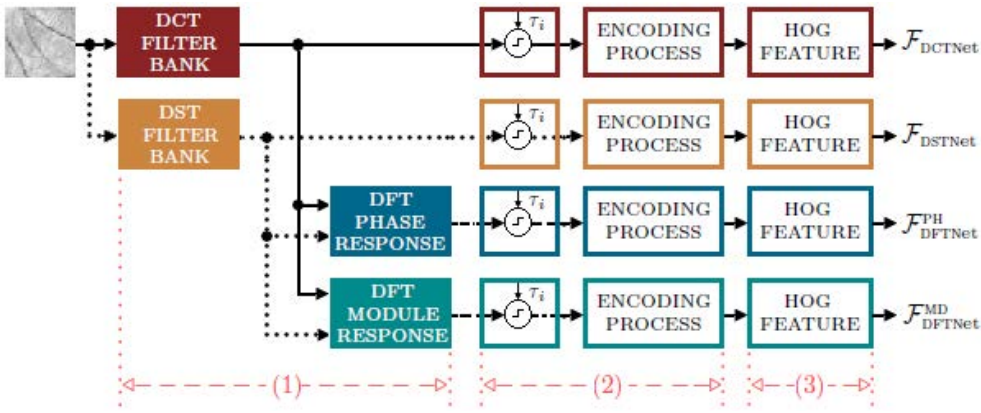

Figure 1. Synopsis of proposed approach outline

1) Analysis step: This is the most important step where an image is passed to a filtering operation by using a group of filters at several stages. This method uses two sets of filters that are extracted with DFT (W_{ij}^{DCT}, W_{ij}^{DST}). Depending on the set that is used, a different vector may be produced. The filter set that is extracted by DCT (W_{ij}^{DCT}) produces vector F_{DCTNet}, while the filter set that is extracted by DST (W_{ij}^{DST}) produces vector F_{DSTNet}. The combination of the filtered images by these two sets can generate two other vectors. Indeed, the combination by the phase gives vector F_{DFTNet}^{PH}, while the combination by the module gives vector F_{DFTNet}^{MD}.

2) Pooling step: The objective of this step is to reduce the data size while preserving its discrimination. Here, the thresholding principle is used to convert filtered images into binary images, which are then combined by an encoding process. It should be noted that, in our proposed scheme, threshold set (τ_i) can be optimized in the training phase of the feature-extraction method in order to maximize the recognition rate (for a further explanation, see below).

3) Observation vector step: The feature vector is finally produced in the last step. This step receives a set of descriptors as input; to extract feature vector ($\mathcal{V}_i$), a handcrafted feature extraction method is applied to each input. Then, all of the resulting 1D vectors ($\mathcal{V}_i|_{i=1}^{\ell}$) are concatenated into a single flattened feature vector. In our method, an efficient method is used to extract the feature of each descriptor (called a histogram of oriented gradients [HOG]); this is applied on the basis of a block analysis in order to be more efficient.

In fact, the proposed method is based on DCTNet, which uses untrained filters that are produced directly from DCT basis functions [25]. The system that is based on these filters is a simple deep-learning system; its simplicity lies in the fact that the weights of the filters are not adjusted during the training (learning) phase (i.e., the feature-extraction method is independent of the classifier and can produce learning-free deep features – see Fig. 2).

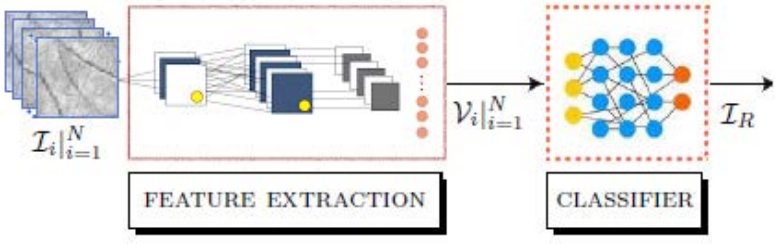

Figure 2. Learning-free deep feature-based classification system

Learning-free deep features often give a low classification rate and can only be improved by changing the structure of the feature-extraction method (*e.g.,* the number of convolution stages, the number and sizes of filters, etc.). In order to adapt the

feature-extraction method with the classifier, an optimization method must therefore be used in order to select the best parameters that can produce learned deep features (see Fig. 3).

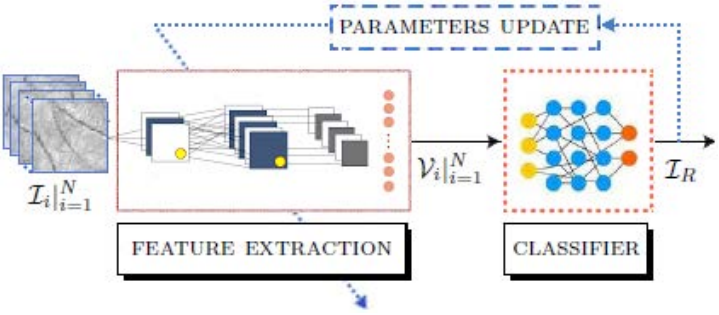

Figure 3. Learned deep feature-based classification system

In the proposed method, the filter weights are not adjusted; we instead optimize the binarization thresholds (τ_i) to produce discriminant vectors that are capable of improving the recognition rate. In addition, the parameters of the extraction method that are applied to the descriptors (in our case, the HOG parameters – including the bin histogram and window) can also be optimized.

3.2. Proposed feature-extraction method

Despite the many ways to implement handcrafted feature-extraction methods, these have unfortunately reached their accuracy limits when representing these features. Recently, relevant research has been devoted to extracting deep features using so-called deep feature-extraction methods. In these methods, an image can be represented with multilevel features from which we can extract features that cannot be obtained by handcrafted methods. So, these methods are generally based on convolution filters at several stages (layers). Several convolutional deep-network architectures have been proposed for biometric recognition; however, most conventional architectures require extensive training data, resulting in long training times in order to attain acceptable results. Indeed, these requirements do not support the use of these methods in a vast range of applications. Also, these methods suffer from limited memory capacity and slow CPU usage, which inevitably leads to finding simple, fast, and easy deep-learning techniques that can be exploited in most devices that are currently available. As an alternative to these methods, researchers have thus introduced architectures that are based on learning-free filters like the DCTNet architecture. In addition to the set of filters that are formed by DCT (W_{ij}^{DCT}), our scheme also uses a set of filters that are formed by DST (W_{ij}^{DST}). So, we will first explain how these filter sets are formed.

3.2.1. Theoretical prerequisites

This section presents the preliminary requirements (essential background) that are used in the proposed feature-extraction method.

a) Discrete Fourier transform: In the signal and image-processing field, the discrete Fourier transform [32] is a known process for allowing the switching from a signal spatial representation to a spectral one. The DFT of an image decorrelates the pixels of an image, while the information is concentrated in low-frequency coefficients. This transformation contains two parts: the real, and the imaginary (which are represented by the cosine [DCT] and the sine [DST], respectively). Moreover, we can use these two parts to extract information about the amplitude of a signal and its phase. The 1D-DCT and 1D-DST transformation matrices for a square input of a size of B × B are given by the following:

$$\varphi_{ij}^{\Psi} = \begin{cases} \frac{1}{\sqrt{B}} & 1 = 0, 0 \leq j \leq B-1 \\ & 1 \leq i \leq B-1, \\ \sqrt{\frac{2}{B}}\Psi\left(\frac{\pi\,(2j+1)\,i}{2B}\right) & 0 \leq j \leq B-1 \end{cases} \tag{1}$$

where Ψ denotes $\cos(\cdot)$ or $\sin(\cdot)$, and the 2D-DCT is only a product of a vertical base and a horizontal base of the 1D-DCT. Also, a matrix of a size of B × B is generated by utilizing Equation (1), which is used to create the bases of a DCT (or DST) of a size of $\rho \times \rho\,(\rho = B \cdot B)$:

$$\varphi = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1B} \\ a_{21} & a_{22} & \cdots & a_{2B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{B1} & a_{B2} & \cdots & a_{BB} \end{pmatrix} = \begin{pmatrix} V_1 \\ V_2 \\ \vdots \\ V_B \end{pmatrix} \in \mathbb{R}^{B \times B} \tag{2}$$

For creating a different basis in the same matrix, the elements of each row are utilized as weights for all of the rows (as is shown in the following equation):

$$M_k = v_1^T . v_j \in \mathbb{R}^{B \times B}, \quad i, j = 1, 2, \cdots, B, \quad k = 1, 2, \cdots, \rho \tag{3}$$

After this, each resulting matrix M_k is reorganized into a 1D vector (V_k):

$$V_k = F_{\rho \times 1}^T(M_k) \in \mathbb{R}^{\rho \times 1} \tag{4}$$

where $F_{\rho \times 1}(\cdot)$ is a function that maps matrix $M_k \in \mathbb{R}^{B \times B}$ to a vector $V_k \in \mathbb{R}^{\rho \times 1}$. The obtained $V_k|_{k=1}^{\rho}$ vectors are then concatenated into a single vector(V):

$$V = [V_1, V_2, \cdots, V_\rho] \in \mathbb{R}^{\rho \times \rho} \tag{5}$$

Now, for rearranging the elements of this vector (V), the zig-zag (F_z) sweep technique is utilized:

$$\widehat{V} = F_z(V) = [\widetilde{V}_1, \widetilde{V}_2, \cdots, \widetilde{V}_\rho] \in \mathbb{R}^{\rho \times \rho} \tag{6}$$

The zig-zag technique is used to reorganize $\widehat{V}$ by placing the important elements at the beginning of the vector based on their importance (low frequency to high frequency) in which the first vector ($\widetilde{V}_1$ – first column of $\widehat{V}$), called the DC element, represents the mean value.

b) Filter formulation: A few years ago, Chan *et al.* [12] proposed a lightweight deep-learning network called PCANet; after its unexpected success for most image-classification tasks (and because of its very simple structure), Ng *et al.* [31] proposed another structure that was similar to that of Chan *et al.* – this was called DCTNet. The main difference between the two lied in the formulation of the filters. This suggestion is no accident but rather the result of the close relationship between PCA and DCT, which was proven by Chan *et al.* in their work. The main difference between DCT and PCA is that PCA must be defined with respect to a given data set (PCANet filters are, therefore, formulated by a learned method), whereas DCT is absolute and is determined only by the input size and does not need a particular data set (DCTNet filters are, therefore, formulated by a learning-free method). The following is the method of formulating the convolutional filters that was used by the founders of DCTNet. Before this, it should be noted that the sine function is nothing more than the cosine function shifted by $\frac{\pi}{2}$; for this reason, both sets of filters follow the same formulation algorithm.

Let η be the number of filters of a size of $k_1 \times k_2$ that are used at a given stage. From the 2D-DCT vector ($\widehat{V}$), the convolution filters are chosen by using Equation (6). Unlike in the case of a PCANet system, the DC element is not considered to be a filter in a DCTNet or DSTNet system (and, thus, DFTNet); for a better performance, the average of each patch is removed. The selection of the bases, therefore, starts from 2 to $\eta + 1$:

$$V_F = \widehat{V}_k(i)|_{i=2}^{\eta+1}, \in \mathbb{R}^{\rho \times \eta}, \qquad \rho = k_1 \cdot k_2 \tag{7}$$

Finally, a set of filters is obtained as follows:

$$W_j = F_{k_1 \times k_2}[V_F(j)] \in \mathbb{R}^{\rho \times \eta} \qquad j = 1, 2, \cdots, \eta \tag{8}$$

where W_j denotes filter j, and $F_{k_1 \times k_2}$ is a function that maps vector $V_F(j)|_{j=1,2,\cdot,\eta}$ $\in \mathbb{R}^{\rho \times 1}$ to a matrix $W_j \in \mathbb{R}^{k_1 \times k_2}$.

3.2.2. Functional behavior

In Figure 4, we present the proposed structure of our deep feature-extraction method (DFTNet deep learning), which consists of two stages. The approved structure could be split into five main layers: convolutional layer (two stages), the DFT response calculation layer ((real part (DCT), imaginary part (DST)), amplitude or phase), binarization layer, hash layer, and the feature vector layer (block-based HOG).

For the system framework description, the input images are assumed to be of a size of $H \times W$, and the patch size (*i.e.,* the size of the convolutional filter [from DCT and DST]) for stage ℓ is as follows:

$$W_i^\ell = k_1^\ell \times k_2^\ell, \qquad i \in [1, \cdots, L_\ell], \qquad \ell \in [1, \cdots, S_\ell] \tag{9}$$

where L_ℓ denotes the number of filters in convolution layer ℓ and is the number of convolution layers (stages). It should be noted that $K_j^\ell|_{j=1,2}$ is an odd integer that satisfies the $K_j^\ell \leq H$ and $K_j^\ell \leq W$ conditions.

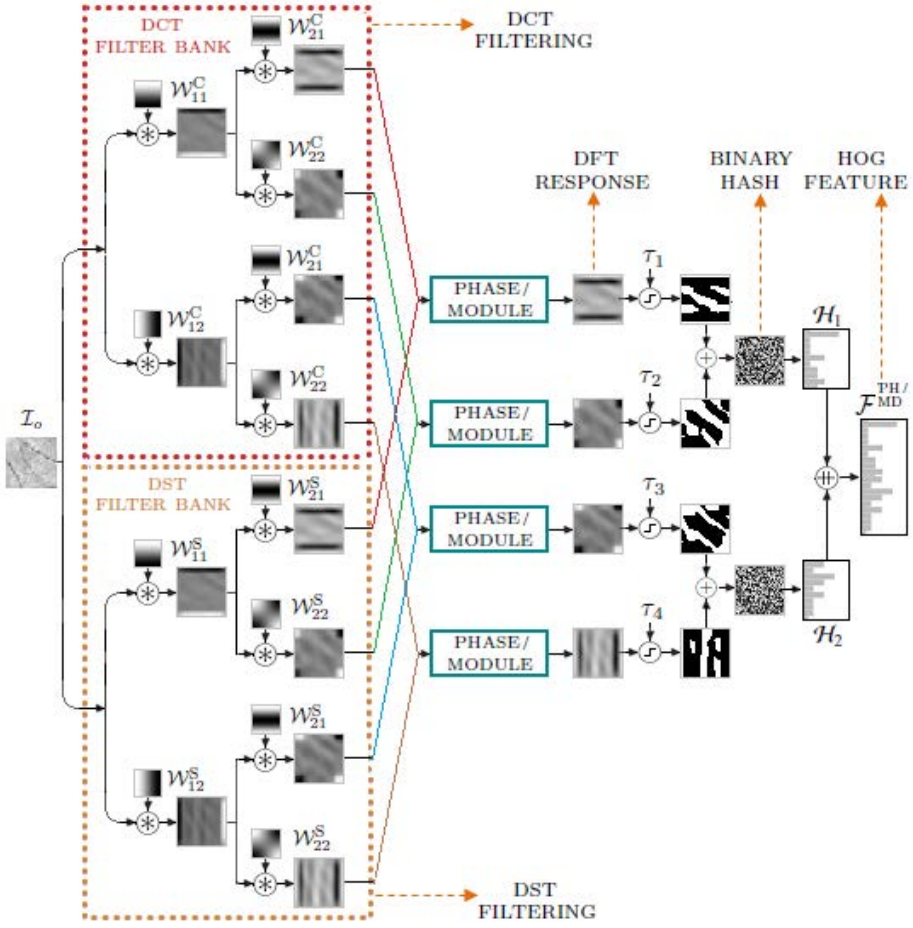

Figure 4. Proposed learning-free deep features for multispectral palm print-classification scheme (DFT[MD/PH]Net) – example of two stages with two convolution filters in each stage

a) Convolutional layer. The DFTNet deep feature-extraction method is a variant of convolutional-based deep-learning techniques, which based on the size and the number of filters, and at several different stages with several filters the input image convoluted, and after passing it to the pooling layer, it outputs the feature vector. Note that, unlike PCANet, DFTNet does not include a learning phase to train the convolution filters because it directly uses the DCT and DST bases as filters. It should also be noted that our method (DFTNet) consists of two completely similar methods

(DCTNet and DSTNet), which differ only in the type of filter. For practical reasons, we are going to explain only one method – DCTNet.

• **Stage 1.** This stage is about creating L_1 convolution filters by using a resulting vector $\widehat{V}$ of the formed DCT (or DST) bases by using Equation (6) with $B = k_1^\ell = k_2^\ell$:

$$V_F = \widehat{V}(j)|_{j=2}^{L_1+1}, \in \mathbb{R}^{\rho \times L_1}, \qquad \rho = k_1^1 \cdot k_2^1 \tag{10}$$

At this stage, the set of filters is obtained as follows:

$$W_i^1 = F_{k_1^1 \times k_2^1}[V_F(i)] \in \mathbb{R}^{k_1^1 \times k_2^1}, \qquad i = 1, 2, \cdots, L_1 \tag{11}$$

The output (L_1 filtered images) of this stage is given by Equation 12 by filtering the input image (I) using filters $W_i^1|_{i=1}^{L_1}$:

$$I_i^1 = I * W_i^1, \qquad i = 1, 2, \cdots, L_1 \tag{12}$$

where $*$ denotes the 2D convolution process. In order to make the obtained I_i^1 filtered images have the same size as I ($H \times W$), the boundary of I is zero-padded before convolution.

• **Stage 2.** As in the first stage, the same process is repeated on all the resulting filtered images. By using Equation 6 (with $B = k_1^2 = k_2^2$), the basis of DCT (or DST) is formed for later using the resulting vector $\widehat{V}$ to create the L_2 convolution filters:

$$V_F = \widehat{V}(j)|_{j=2}^{L_2+1}, \in \mathbb{R}^{\rho \times L_2}, \qquad \rho = k_1^2 \cdot k_2^2 \tag{13}$$

In this stage, the convolution filters are given as follows:

$$W_j^2 = F_{k_1^2 \times k_2^2}[V_F(j)] \in \mathbb{R}^{k_1^2 \times k_2^2}, \qquad i = 1, 2, \cdots, L_2 \tag{14}$$

Then, we get the outputs of the second stage by filtering the input images ($I_i^1|_{i=1}^{L_1}$) by using the $W_j^2|_{j=1}^{L_2}$ filters:

$$I_{i,j}^2 = I_i * W_j^2, i = 1, 2, \cdots, L_1, j = 1, 2, \cdots, L_2 \tag{15}$$

Finally, the output of this stage is $L_1 \cdot L_2$ filtered images for each input image and, thus, by obtaining L_2 filtered images for each filtered image from the first stage.

b) DFT Response Layer. In fact, our system contains four subsystems (including the original one [DCTNet]); it can be configured in this layer to work as one of these systems: DCTNet, DSTNet, DFT$^{\text{MD}}$Net, or DFT$^{\text{PH}}$Net. Thus, the filtering results of the last stage $I_{i,j}^2|_{i=1,2,\cdots,L_1,j=1,2,\cdots,L_2}^{DCT}$ DCT (obtained from the DCT filters) and $I_{i,j}^2|_{i=1,2,\cdots,L_1,j=1,2,\cdots,L_2}^{DST}$ DST (obtained from the DST filters) represent the real and imaginary parts of the DFTNet system, respectively. These two responses are combined to produce the phase and the amplitude as follows:

$$\text{DCTNet:} \quad I_{ij}^{\text{Re}} = I_{ij}^2|_{i=1,2,\cdots,L_1,j=1,2,\cdots,L_2}^{\text{DCT}} \tag{16}$$

$$\text{DSTNet:} \quad I_{ij}^{\text{Im}} = I_{ij}^{2}|_{i=1,2,\cdots,L_1,j=1,2,\cdots,L_2}^{\text{DST}} \tag{17}$$

$$\text{DFT}^{\text{PH}}\text{Net:} \quad I_{ij}^{\text{Ph}} = I_{ij}^{2}|_{i=1,\cdots,L_1,j=1,\cdots,L_2}^{\text{DCT-DST}}, \quad I_{ij}^{\text{Ph}} = \arctan(\frac{I_{ij}^{2\text{DST}}}{I_{ij}^{2\text{DCT}}}) \tag{18}$$

$$\text{DFT}^{\text{MD}}\text{Net:} \quad I_{ij}^{\text{Md}} = I_{ij}^{2}|_{i=1,\cdots,L_1,j=1,\cdots,L_2}^{\text{DCT-DST}}, \quad I_{i,j}^{\text{Md}} = \sqrt{(I_{ij}^{2\text{DCT}})^2 + (I_{ij}^{2\text{DST}})^2} \tag{19}$$

c) Binarization Layer. The process of binarization (binary quantization) transforms the obtained real values into a binary format; then, a threshold τ_b to zero is applied (Equation 20) for both the real and imaginary parts, as the response values have the same probability of being negative or positive:

$$I_{ij}^{b}(n,m) = \begin{cases} 1 & \text{if} \quad I_{ij}^{\text{X}}(n,m) \geq \tau_b \quad \text{X} \equiv \{\text{Re, Im,Md,Ph}\} \\ 0 & \text{if} \quad I_{ij}^{\text{X}}(n,m) < \tau_b \quad i = 1,\cdots,L_1, \quad j = 1,\cdots,L_2 \end{cases} \tag{20}$$

It should be noted that, as the phase (PH) sign always follows the sign of the imaginary part (DST), a threshold that is equal to zero produces the same binarization results for both parts. On the other hand, the components of the amplitude (MD) are positive, so a threshold of zero gives a result in which all of the components are equal to 1 (white image). Therefore, the binarization threshold for the amplitude (MD) and phase (PH) is determined as follows:

$$\tau_b = k \cdot \varrho_{\text{X}}, \quad \text{X} \equiv \{\text{Md, Ph}\} \tag{21}$$

where ϱ_{X} is the mean value of the modulus (or of the phase), and k is a predefined factor that is varied in interval $[0.05,\cdots,1]$ with a step of 0.05.

d) Hash Layer. For the purpose of reducing the amount of data, this layer converts the L_2 binarized images into a single integer-valued "image." Thus, using the next decoding polynomial, a conversion (binary to decimal) of the binary code (encoded on L_2 bits) around each pixel is applied:

$$I_j^3 = \Sigma_{j=0}^{L_2-1} I_{ij}^{b} \cdot 2^j, \quad i = 1,\cdots,L_1 \tag{22}$$

In DCTNet (like in PCANet), the output number of this layer is the same as the filter number that is utilized in the first stage of the convolution. So, a separate decoding of each L_2 group gives an images set that is equal to L_1.

e) Feature Vector Layer. In this layer, the feature size of each image is reduced. For each image among the L_1 images, the histogram is then calculated to form the feature vector, and all of the obtained histograms are concatenated. Unlike PCANet and the original DCTNet, our proposed method (DFTNet) uses an oriented gradient histogram (HOG) [1]. Now, to get the feature vector of each input image, we fist divide each $I_j^3|_{i=1}^{L^1}$ image into N_b blocks(B) as follows:

$$N_b = \left\lfloor \frac{H - b_1}{o} + 1 \right\rfloor \cdot \left\lfloor \frac{W - b_2}{o} + 1 \right\rfloor \tag{23}$$

where o denotes the horizontal/vertical overlap between two adjacent blocks, $b_1 \times b_2$ is the analysis block size, and $\lfloor \cdot \rfloor$ is the integer part of the value. Then, a set of ϕ_i blocks for each $(I_j^3|_{i=1}^{L_1})$ image is obtained and defined as follows:

$$\phi_i = \{B_1^i, B_2^i, \ldots, B_{N_b}^i\} \in \mathbb{R}^{(b_1 \times b_2) \times N_b}, \quad B_i^j|_{j=1}^{N_b} \in \mathbb{R}^{b_1 \times b_2}, \quad i = 1, \cdots, L_1 \tag{24}$$

where B_i^j denotes the j^{th} block of image I_i^3. Then, a HOG histogram for each block (B_i^j) is calculated. To form the final feature vector for the examined image, all of the resulting vectors from the L_1 images are concatenated.

In the HOG technique, the input image is analyzed by window (W_{HOG}) in which each window is divided into non-overlapping cells. For each pixel of each cell, the orientation is then calculated to form the histogram, and the magnitude of the gradient is calculated and used as the voting weight. Finally, a concatenation of the set of histograms of the cells is done in each window to form the HOG descriptor. So, each block feature is extracted using the HOG technique as follows:

$$H_j^i = F_{\text{HOG}}(B_j^i) \in \mathbb{R}^{\lambda \times 1}, \quad j = 1, 2, \cdots, N_b, \quad i = 1, 2, \cdots, L_1 \tag{25}$$

where F_{HOG} denotes the HOG feature-extraction process, and λ is the block histogram length. Note that the (λ) value is a function of the number of both the HOG windows (η_ω) and the histogram bins (η_b). For each image $((I_j^3|_{i=1}^{L_1}))$, all of the HOG vectors (features extracted) from all of the blocks are then concatenated into a single vector (ϑ_i):

$$\vartheta_i = \left[H_1^i, H_2^i, \ldots, H_{N_b}^i\right] \in \mathbb{R}^{(N_b \lambda) \times 1}, \quad i = 1, 2, \cdots, L_1 \tag{26}$$

Finally, the feature vector of the input image is obtained as follows:

$$V_T = [\vartheta_1, \vartheta_2, \ldots, \vartheta_{L_1}] \in \mathbb{R}^{(N_b \lambda L_1) \times 1} \tag{27}$$

It is important to note that, depending on the block size $(b_1 \times b_2)$ and the overlap rate (o), the length and precision of the (V_T) vector of each input image are changed.

4. Experimental results

In this section, a detailed performance evaluation of the proposed method is presented. In our experiments, CASIA's multispectral palm-print database [10] of 100 people was used to test the proposed method in a biometric-identification system. In this database, 12 images for each person were acquired using a low-resolution contactless sensor as grayscale format and collected in two separate sessions with a gap of more than a month (with captures of 6 images in each session). The data set was divided into two galleries; a random selection of three samples for each person was applied and assigned for the enrollment phase (training), and the rest of the samples were used for recognition phase (test). A matching score of 45,450 was obtained when using all of the test images divided into two experimental classes; the genuine ones with a score of 900, and the impostors with 44,550 scores.

Several experiments were carried out in this work that can be divided into two main parts. In the first part, the efficiency of the four proposed systems (DCTNet, DSTNet, DFT^{MD}Net, and DFT^{PH}Net) was evaluated by selecting their optimal parameters in order to choose the best system among them; the second part compared the efficiency of the best system to systems that were based on traditional (hand-crafted) and learned (learned hand-crafted) feature-extraction methods.

4.1. System performance

The objective of this part was to test the proposed biometric systems. It was also divided into two sub-parts: the first one selected the optimal parameters for each system by dealing with the preliminary experiments, while the second focused on selecting the best proposed system.

4.1.1. Preliminary experiments

In biometrics, good choices of features play a major role in obtaining an exact result for a person-recognition system, as it conditions the performance of any recognition methodology. For the best performance evaluation of an identification system, it is essential to select the appropriate parameters that can give the best results. Mathematically, a direct formula for finding out the optimal parameters that provide the best performance does not exist. In this work, the experimental tests were carried out by varying the different parameters, including the number of filters in each stage ($L_i|_{i=1}^{2}$) and the size of the filters in each stage ($k_i|_{i=1}^{2}$) in a predefined set of values in order to control the precision of the extracted feature vectors by selecting the combination that optimizes an objective function (empirical tests). First of all, it should be noted that our tests were carried out on an open-set biometric identification system based on the "WHITE" palm print spectral band. In addition, the system was developed and implemented using Matlab 2009 on a Windows 7 platform and an embedded PC with a 2.2 GHz Intel Pentium processor and a DRAM of 2 GB.

The performance of the proposed biometric system was evaluated according to the four applied feature-extraction methods (DCTNet, DSTNet, DFT^{MD}Net, and DFT^{PH}Net) by varying one parameter each time. Under the first stage ($\ell = 1$), the parameters were chosen from a predefined set of values (2, 4, 6, 8, and 10) for the number of filters (L_1) and (9×9, 11×11, 13×13, 15×15, 17×17) for the sizes of the L_1 convolution filters ($k_1 \times k_2$). For the DFT^{PH}Net and DFT^{MD}Net systems, the binarization threshold parameter (κ) was preselected to 0.5. It should be noted that our systems used the SVM classifier due to its well-demonstrated effectiveness in many classification applications [30].

In an open-set biometric identification system, an identification rate (genuine acceptance rate – GAR) is calculated for each feature-extraction method in order to examine the impact of these parameters on the system's performance. The test results of the proposed system in a single convolutional stage are shown in Figure 5 and can be illustrated in the following points: i) all possible combinations of L_1 and

($k_1 \times k_2$) can show very acceptable performance with an open-set identification rate GAR that is greater than 99.20%; ii) the curves show how the number of filters and the identification rate are proportional (best case with ten filters) and how a large filter can improve the system's performance; and iii) the best system performance can be achieved using the DFT$^{\mathrm{MD}}$Net method.

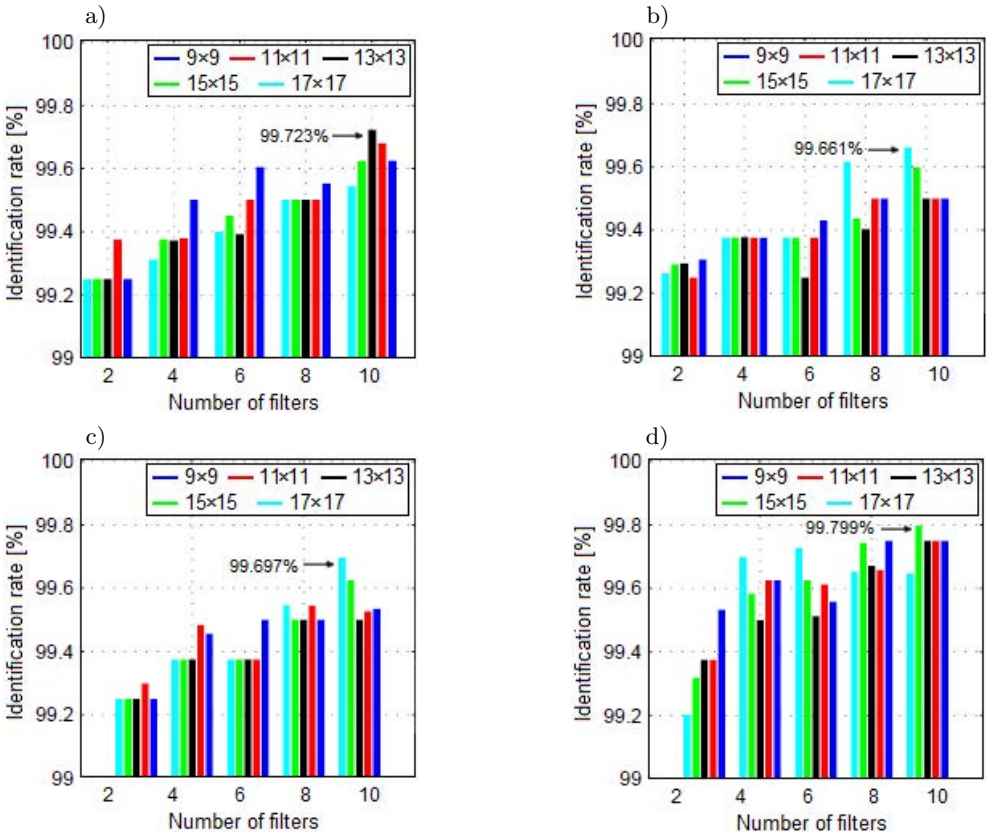

Figure 5. Biometric system performance with single stage: a) biometric system based on DCTNet; b) biometric system based on DSTNet; c) biometric system based on DFT$^{\mathrm{PH}}$Net; d) biometric system based on DFT$^{\mathrm{MD}}$Net

According to Figure 5, it is clear that ten convolution filters provided better results in terms of GAR. In this case, the system could reach an error-identification rate (equal error rate – EER) that was equal to 0.2767% at a threshold T_o of 0.6396, 0.3391% at $T_o = 0.6296$, and 0.3034% at $T_o = 0.6283$ by using filter sizes of 13×13, 17×17, and 17×17 for the DCTNet-, DSTNet-, and DFT$^{\mathrm{PH}}$Net-based systems, respectively, with improvements of 27.20, 40.58, and 33.45%. Using the DFT$^{\mathrm{MD}}$Net method with filter size of 15×15, the system attained an EER of 0.2015% at $T_o = 0.7068$. Although acceptable results can be obtained with one convolutional stage,

recognition errors could happen. Fortunately, our system can be configured in multiple stages, which can reduce this and improve the system's efficiency. For this reason, a re-examination of the systems that have been proposed under two stages of convolution was done in order to improve their performance. Therefore, the same test methodology as in the previous part was carried out with all of the proposed systems using four convolution filters of a size of 15 × 15 in the first stage. The test results of the efficiency of this configuration in the open-set identification mode for all of the biometric systems are presented in Figure 6.

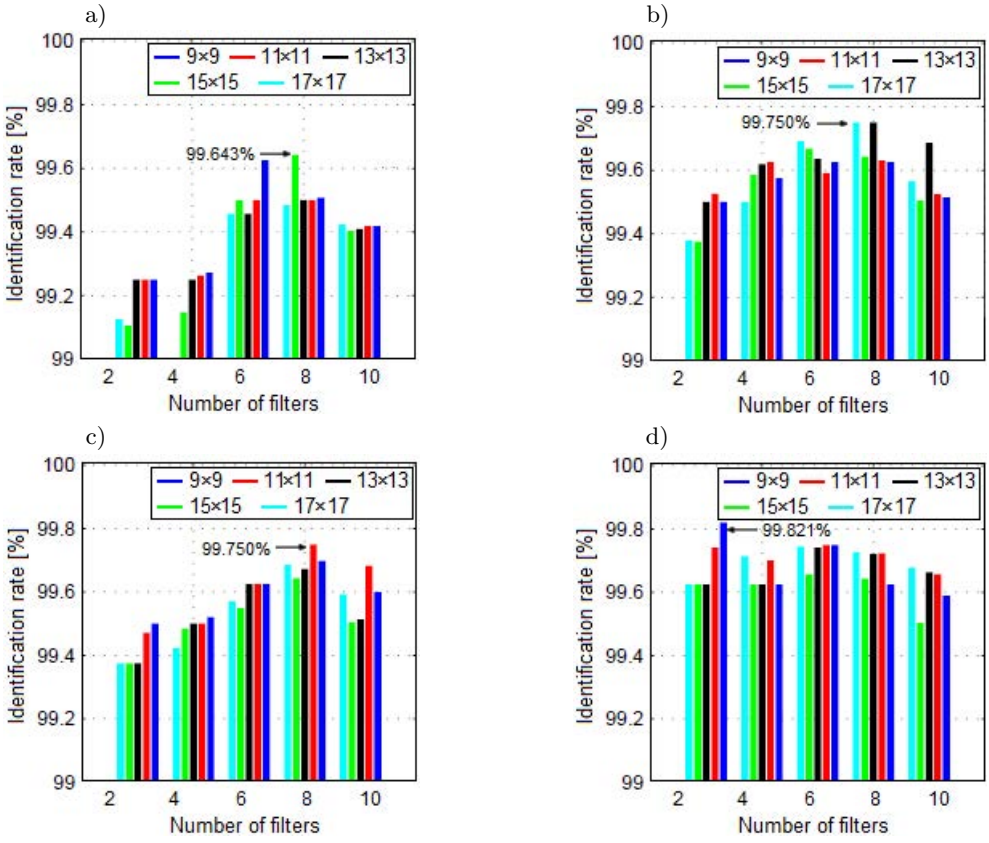

Figure 6. Biometric system performance using two stages: a) biometric system based on DCTNet; b) biometric system based on DSTNet; c) biometric system based on DFT^{PH}Net; d) biometric system based on DFT^{MD}Net

Also, two important remarks could be underlined from these tests: the use of two stages of convolution improves the proposed biometric system's performance, and when the number of convolution stages is high, the better the small filters are (which will give better results) – especially if the sizes of the filters in the first convo-

lution stages are large. Through Figure 6, we notice that performance improvements of 26.25, 22.12, and 10.95% were achieved for those systems that were based on DSTNet, DFT^{PH}Net, and DFT^{MD}Net, respectively. On the other hand, a performance degradation of 57.27% was obtained for the DCTNet case. In addition, the best performance was obtained when the using DFT^{MD}Net method, which gave a best result of EER (0.179%) at a threshold of $T_o = 0.7003$. In general, these enhancements were very acceptable, and their importance increased with the growth of the database. Other good configurations can also be obtained with EERs of 0.250% at $T_o = 0.6138$ and $T_o = 0.6170$ in the case of DSTNet and DFT^{PH}Net, respectively, and an EER of 0.357 at $T_o = 0.5966$ when using the DCTNet method. From these results, we chose to implement the DFT^{MD}Net method to evaluate the proposed identification system in both open-set and closed-set modes by first choosing the best binarization threshold (κ) from the [0.05, ..., 1.00] interval.

4.1.2. Biometric system performance

For a performance evaluation of the proposed biometric system that was based on DFT^{MD}Net, the system was tested with two convolutional stages to be classified later by using the KNN and SVM classifiers. In our tests, an appropriate contrast for image-feature representation could be obtained by using a wide range of binarization threshold (κ) choices ($\kappa \in [0.05, \ \ldots, \ 1]$) with a step of 0.05, which gave 20 cases. Consequently, we can choose the suitable number with texture analysis in a specific application. To show the impact of the threshold binarization on the biometric system's performance, the best open-set identification test results in terms of EER are shown in Table 2.

Table 2

DFT^{MD}Net biometric system performance under different spectral sub-bands

Stage	Band [nm]	KNN			SVM		
		κ	T_o	EER	κ	T_o	EER
1	WHITE	0.60	0.2505	0.250	**0.60**	**0.7695**	**0.125**
	460	0.30	0.1935	0.125	**0.40**	**0.6883**	**0.125**
	630	0.25	0.2895	0.250	0.30	0.6275	0.250
	700	0.20	0.2662	0.410	0.45	0.7335	0.250
	850	0.20	0.3084	0.875	0.25	0.6687	0.875
	940	0.25	0.3033	1.000	0.20	0.6360	1.250
2	WHITE	0.30	0.2340	0.250	0.45	0.6973	0.178
	460	0.30	0.2171	0.250	0.45	0.7255	0.125
	630	**0.30**	**0.2144**	**0.193**	0.20	0.7080	0.250
	700	0.15	0.1487	0.180	**0.30**	**0.7309**	**0.080**
	850	0.15	0.2447	0.804	**0.20**	**0.7185**	**0.689**
	940	0.20	0.2709	0.963	**0.25**	**0.6593**	**0.875**

From the results that are displayed in Table 2, it can be observed that the best results were obtained by using the SVM classifier. In the case of one convolutional

stage, a best EER of 0.125% was achieved at the thresholds of $T_o = 0.7695$ and $T_o = 0.6883$ with binarization thresholds of 0.6 and 0.4 when using the WHITE and 460-nm spectral bands, respectively. When using two convolutional stages, a best EER of 0.080% was obtained at a threshold of $T_o = 0.7309$ with a binarization threshold of 0.30 when using the 700-nm spectral band; this shows the efficiency of the SVM classifier as compared to KNN in terms of identification rate and execution speed.

To be certain of our method, Table 3 shows the performance evaluation under the two identification modes; in this, we notice that, when using the SVM classifier the 700-nm spectral band gave the best results in both identification modes, with a minimum EER of 0.080 ($T_o = 0.7309$, $\kappa = 0.3$) in open-set mode as well as an identification rate (rank-one recognition – ROR) of 99.250%, with a minimum rank of perfect recognition (RPR) of 10 being achieved in the closed-set mode.

Table 3
DFT^{MD}Net biometric system performance for each spectral band in both identification (open-set/closed-set) modes

Band	κ	Open-set		Closed-set		Classifier	Stage
		T_o	EER	RPR	ROR		
WHITE	0.60	0.6973	0.125	13	98.375	SVM	1
460	0.40	0.6882	0.125	60	99.250	SVM	1
630	0.30	0.2144	0.193	71	98.875	KNN	2
700	0.30	0.7309	**0.080**	10	**99.250**	SVM	2
850	0.20	0.7185	0.689	45	96.625	SVM	2
940	0.25	0.6593	0.875	81	97.000	SVM	2

4.2. Comparative study

To prove the efficiency of our method, two important comparisons were made: one to demonstrate the efficiency of deep feature-extraction methods as compared to those that are based on classical methods, and the second to demonstrate the efficiency of our method as compared to the recent methods that are based on deep learning. It should be noted that the comparison was limited to open-set biometric identification systems based only on 850-nm, 940-nm, and WHITE spectral bands, which were sufficient to prove the efficiency of our feature-extraction method.

4.2.1. Comparison to classical methods

As our method is intended for image texture analysis, we will try to examine the efficiency of the proposed deep feature-extraction method in this subpart as compared to classical methods based on local binary pattern (LBP), including hand-crafted methods (three-patch LBP – TP-LBP) and four-patch LBP – FP-LBP) [39] and a learned hand-crafted method (compact binary descriptor – CBD) [28]. For a fair comparison, three important conditions must be considered. First, feature-extraction methods should be performed with the same testing methodology. Second, all methods must

be performed on the same implementation platform. Third, all methods must be implemented on the same machine. These conditions are taken later into account to help explain the performance effectiveness. In order to adapt the CBD method to the database that was used and to also improve the system's performance, an additional test of the size of the pixel neighborhood (R) and the length of the projection matrix (BIN) was applied. The obtained test results for the three spectral bands according to the previously mentioned points are illustrated in Table 4.

Table 4
Selection of CBD parameters under three spectral bands

Band	Windows = 15×15 ($R = 7$)					
	BIN = 19		BIN = 23		BIN = 25	
	T_o	EER	T_o	EER	T_o	EER
850	0.614	1.298	0.658	1.196	0.684	1.018
940	0.705	0.485	0.655	0.678	0.674	0.750
WHITE	0.712	1.157	0.698	0.875	**0.726**	**0.250**
Band	Windows = 19×19 ($R = 9$)					
	BIN = 19		BIN = 23		BIN = 25	
	T_o	EER	T_o	EER	T_o	EER
850	0.619	1.307	0.617	1.375	**0.654**	**0.994**
940	0.659	0.500	**0.700**	**0.409**	0.653	0.625
WHITE	0.688	1.000	0.736	0.921	0.711	0.875

From this table, we can clearly see that both of the CBD parameters can give good results in all spectral bands, whereas the best identification rate EER of 0.250 at $T_o = 0.726$ was obtained when using the WHITE spectral band in the case of $R = 7$ with BIN = 25. In the case of $R = 9$, a best case of an EER of 0.409% was obtained at $T_o = 0.700$ with a BIN of 25 when using the 940-nm spectral band with BIN = 25 the highest EER (EER = 0.994% at $T_o = 0.654$) did not exceed 100% (850-nm spectral band). It should be noted that, in all of these tests, the regions that were numbered within a image of 1 to 16 and a codebook size from 25 to 500 (with a step of 25) were also tested. The obtained results showed that 9 regions with a codebook size of 100 was sufficient for better performance.

The second set of experiments aimed to evaluate the system performance by comparing our proposed method with the CBD method (using the optimal parameters that were previously selected) and the two LBP variant-based methods (TP-LBP and FP-LBP). The test results for the three spectral bands in both identification modes are shown in Table 5.

From this table, we observe that, in the open-set identification mode, the best case for the three 850-nm, 940-nm, and WHITE spectral bands were obtained with EER = 0.689% at $T_o = 0.719$, EER = 0.406% at $T_o = 0.700$, and EER = 0.125% at $T_o = 0.697$

using the DFT^{MD}Net, CBD, and DFT^{MD}Net methods, respectively; the best case was achieved when using the proposed DFT^{MD}Net methods with EER = 0.125%. In the closed-set identification mode, the best case for the three 850-nm, 940-nm, and WHITE spectral bands were achieved with ROR = 97.000% with RPR = 72, ROR = 98.500% with RPR = 93, and ROR = 98.375% with RPR = 13 using the CBD, CBD, and DFT^{MD}Net methods, respectively; the best case was achieved when using the CBD method with ROR = 98.500%.

Table 5
Performance comparison of DFT^{MD}Net with classical methods

Methods	Open-set Identification					
	850 [nm]		940 [nm]		WHITE	
	T_o	EER	T_o	EER	T_o	EER
FPLBP	0.257	6.798	0.296	4.750	0.245	1.625
TPLBP	0.265	3.250	0.248	2.936	0.206	1.125
CBD	0.654	0.994	**0.700**	**0.406**	0.726	0.250
Proposed	**0.719**	**0.689**	0.659	0.875	**0.697**	**0.125**
Methods	Closed-set Identification					
	850 [nm]		940 [nm]		WHITE	
	ROR	RPR	ROR	RPR	ROR	RPR
FPLBP	71.505	92	81.124	89	93.500	65
TPLBP	86.875	90	87.375	85	94.875	69
CBD	**97.000**	**72**	**98.500**	**93**	98.025	44
Proposed	96.625	45	97.000	81	**98.375**	**13**

By analyzing the results of this table, we can draw two points: methods that are based on learning give better results than other methods (without learning); also, deep learning is generally efficient when compared to other methods – especially if we use several levels of image analysis (several convolutional stages). The obtained experimental results clearly demonstrate the learning principle efficiency in a feature-extraction task where the effectiveness of deep learning becomes evident in large databases, which make these kinds of systems go well with many high-security applications.

4.2.2. Comparison with recent works

In order to show the effectiveness of our proposed method as compared to existing methods (handcrafted/based deep learning), a comparative study with some recent works is carried out in this part using the CASIA multispectral database, where the main works are summarized in Table 6.

The displayed results in this table shows that the proposed method gave a good performance (with an EER of 0.0800% and an accuracy of 99.25%) as compared to

many other related works that have been carried out on CASIA's multispectral palm-print database. Based on all of the above, we believe that this work results in a new and recent technique that allows us to obtain the best identification performance in all bands of the CASIA palm-print database; in general, it gives an EER that is greater than it was achieved by the others. In addition, our palm print-identification system can be improved by modifying the feature-extraction process (for example, by increasing the number of DFT^{MD}Net stages to effectively remove the unnecessary features).

Table 6
Performance comparison of DFT^{MD}Net with state-of-the-art methods

Methods	Year	Acronym	Classifier	EER [%]
Classical Methods (Handcrafted)	2010 [14]	SAX	KNN	0.9000
	2011 [27]	Contour-code	Hash Table	0.3000
	2014 [26]	RootSIFT	Hierarchique matching	1.0000
Deep Learning Methods	2016 [13]	Sparse	SRC-SVM	0.2800
	2016 [42]	DoN	KNN	0.5300
	2016 [38]	AlexNet	Hausdroff	0.0803
	2016 [34]	Siamese Net	SVM	1.8600
	2017 [29]	PCANet	SVM	0.1200
	2018 [20]	VGG	Softmax	3.7800
	2019 [22]	PalmNet	KNN	0.7200
	2021 [24]	GPWLD	Optimized AE + Cosine similarity	0.2400
Proposed	–	DFT^{MD}Net	SVM	**0.0800**

5. Conclusion

The goal of any pattern-recognition system is to achieve high accuracy; this mainly depends on a good representation of pattern features. In a biometric-identification system, the main impact of the feature-extraction step on accuracy prompted us to consider developing a new feature-extraction method based on the convolutional deep-learning principle. The main objective of this study was to demonstrate the importance of deep-learning principles in multispectral palm print-identification systems. The experimental results that were obtained when using several spectral bands from the CASIA database showed that our method provided high accuracy as compared to many hand-crafted methods and some deep learning-based methods in the literature. Moreover, the excellent performance that was obtained by using a small number of convolutional layers (two convolutional stages) and convolutional filters (four convolutional filters) proved the lower hardware cost. Also, these experiments demonstrated that our method is very flexible and can be used in any biometric application by selecting the appropriate parameters depending on the desired degree of security. And despite the small size of the palm-print image, our feature-extraction

method gave high performance; this small size is undoubtedly proof of the low processing time and, therefore, the possibility of using it in real-time applications. Despite the effectiveness of the feature-extraction method in biometric systems, these systems may unfortunately be exposed to new security and privacy risks. Therefore, our future work should focus on improving the security of our feature-extraction method to protect the biometric template during transmission and/or storage.

Acknowledgements

The authors are grateful to the anonymous referees for their valuable and helpful comments. This research has been carried out within the PRFU project (Grant: A01L08UN120120180001) of the Department of Electrical Engineering, University of Larbi Tebessi, Tebessa. The authors thank the staff of the LAMIS laboratory for their helpful comments and suggestions.

References

[1] Alhindi T.J., Kalra S., Ng K.H., Afrin A., Tizhoosh H.R.: Comparing LBP, HOG and Deep Features for Classification of Histopathology Images. In: *2018 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, 2018.

[2] Ali M.M.M., Yannawar P.L., Gaikwad A.: Multi-Algorithm of Palmprint Recognition System Based on Fusion of Local Binary Pattern and Two-Dimensional Locality Preserving Projection, *Procedia Computer Science*, vol. 115, pp. 482–492, 2017.

[3] Almaghtuf J., Khelifi F., Bouridane A.: Fast and efficient difference of block means code for palmprint recognition, *Machine Vision and Applications*, vol. 31, 51, 2020. doi: 10.1007/s00138-020-01103-3.

[4] Amraoui A., Fakhri Y., Ait Kerroum M.: Multispectral Palmprint Recognition based on Fusion of Local Features, *6th International Conference on Multimedia Computing and Systems (ICMCS), Rabat, Morocco*, pp. 1–6, 2018.

[5] Aounallah A., Bradji L., Bendjenna H.: Is There Still Confidence In Hand-Crafted Feature Extraction Techniques To Use Them In Biometric Systems?, *IEEE International Conference on Recent Advances in Mathematics and Informatics (ICRAMI), Tebessa, Algeria*, pp. 1–6, 2021.

[6] Barra S., De Marsico M., Nappi M., Narducci F., Riccio D.: A Hand-based Biometric System in Visible Light for Mobile Environments, *Information Sciences*, vol. 479, pp. 472–485, 2019.

[7] Bendjenna H., Meraoumia A., Chergui O.: Pattern recognition system: from classical methods to deep learning techniques, *Journal of Electronic Imaging*, vol. 27(3), 2018.

[8] Bensid K., Samai D., Laallam F.Z., Meraoumia A.: Deep learning feature extraction for multispectral palmprint identification, *Journal of Electronic Imaging*, vol. 27(3), 2018.

[9] Bouchemha A., Meraoumia A., Laimeche L., Houam L.: Learning Hand-Crafted Palm-Features for a High-Performance Biometric Systems, *WITS 2020: Proceedings of the 6th International Conference on Wireless Technologies, Embedded, and Intelligent Systems*, pp. 855–866, 2020.

[10] CASIA Multispectral palmprint database, 2005. http://biometrics.idealtest.org/.

[11] Chaa M., Boukezzoula N., Meraoumia A.: Features-Level Fusion of Reflectance and Illumination Images in Finger-Knuckle-Print Identification System, *International Journal on Artificial Intelligence Tools*, vol. 27(3), 2018.

[12] Chan T.H., Jia K., Gao S., Lu J., Zeng Z., Ma Y.: PCANet: A Simple Deep Learning Baseline for Image Classification?, *IEEE Transactions on Image Processing*, vol. 24(12), pp. 5017–5032, 2015.

[13] Charfi N., Trichili H., Alimi A.M., Solaiman B.: Local invariant representation for multi-instance toucheless palmprint identification. In: *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 003522–003527, 2016.

[14] Chen J., Moon Y.S., Wong M.F., Su G.: Palmprint authentication using a symbolic representation of images, *Image and Vision Computing*, vol. 28(3), pp. 343–351, 2010.

[15] Chen Y.-Y., Hsia C.-H., Chen P.-H.: Contactless Multispectral Palm-Vein Recognition With Lightweight Convolutional Neural Network, *IEEE Access*, vol. 9, pp. 149796–149806, 2021.

[16] Chergui O., Bendjenna H., Meraoumia A., Chitroub S.: Combining palmprint and finger-knuckle-print for user identification, *International Conference on Information Technology for Organizations Development (IT4OD), Fez, Morocco*, pp. 1–5, 2016.

[17] Cho S., Oh B.-S., Toh K.A., Lin Z.: Extraction and Cross-Matching of Palm-Vein and Palmprint From the RGB and the NIR Spectrums for Identity Verification, *IEEE Access*, vol. 8, pp. 4005–4021, 2020.

[18] Dvorak M., Drahansky M.: Hand shape recognition and palm-print recognition using 2D and 3D features, *Hand-Based Biometrics: Methods and Technology*, pp. 283–307, 2018.

[19] El-Ghandour M., Obayya M.I., Yousef B., Areed N.F.: Palmvein Recognition using Block-Based WLD Histogram of Gabor Feature Maps and Deep Neural Network with Bayesian Optimization, *IEEE Access*, vol. 9, pp. 97337–97353, 2021.

[20] Fei L., Lu G., Jia W., Teng S., Zhang D.: Feature Extraction Methods for Palmprint Recognition: A Survey and Evaluation, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49(2), pp. 346–363, 2018.

[21] Fei L., Zhang B., Xu Y., Huang D., Jia W., Wen J.: Local Discriminant Direction Binary Pattern for Palmprint Representation and Recognition, *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30(2), pp. 468–481, 2020.

[22] Genovese A., Piuri V., Plataniotis K.N., Scotti F.: PalmNet: Gabor-PCA Convolutional Networks for Touchless Palmprint Recognition, *IEEE Transactions on Information Forensics and Security*, vol. 14(12), pp. 3160–3174, 2019.

[23] Genovese A., Vincenzo P., Fabio S.: *Touchless Palmprint Recognition Systems*, Springer Cham, Switzerland, 2014.

[24] Hassaballah M.H., Khalid M.: Recent Advances in Computer Vision, *Studies in Computational Intelligence*, vol. 804, 2019.

[25] Jing X.Y., Zhang D.: A face and palmprint recognition approach based on discriminant DCT feature extraction, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34(6), pp. 2405–2415, 2004.

[26] Kang W., Liu Y., Wu Q., Yue X.: Contact-free palm-vein recognition based on local invariant features, *PLoS ONE*, vol. 9(5), 2014.

[27] Khan Z., Mian A., Hu Y.: Contour code: Robust and efficient multispectral palmprint encoding for human recognition. In: *International Conference on Computer Vision, Barcelona, Spain*, pp. 1935–1942, 2012.

[28] Lu J., Liong V.E., Zhou X., Zhou J.: Learning Compact Binary Face Descriptor for Face Recognition, *Transactions on Pattern Analysis and Machine Intelligence*, vol. 37(10), pp. 2041–2056, 2015.

[29] Meraoumia A., Kadri F., Bendjenna H., Chitroub S., Bouridane A.: Improving Biometric Identification Performance Using PCANet Deep Learning and Multispectral Palmprint. In: *Biometric Security and Privacy. Signal Processing for Security Technologies*, pp. 51–69, Springer, Cham, 2017.

[30] Michele A., Colin V., Santika D.D.: MobileNet Convolutional Neural Networks and Support Vector Machines for Palmprint Recognition, *Procedia Computer Science*, vol. 157, pp. 110–117, 2019.

[31] Ng C.J., Teoh A.B.J.: DCTNet: A simple Learning-free Approach for Face Recognition, *Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), Hong Kong, China*, pp. 761–768, 2015.

[32] Olson T.: The Discrete Fourier Transform. In: *Applied Fourier Analysis*, Birkhäuser, New York, NY, 2017.

[33] Qin H., El Yacoubi M.A., Lin J., Liu B.: An Iterative Deep Neural Network for Hand-Vein Verification, *IEEE Access*, vol. 7, pp. 34823–34837, 2019.

[34] Svoboda J., Masci J., Bronstein M.M.: Palmprint recognition via discriminative index learning. In: *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 4232–4237, 2016. doi: 10.1109/ICPR.2016.7900298.

[35] Thamri E., Aloui K., Naceur M.: Improving Palmprint based Biometric System Performance using Novel Multispectral Image Fusion Scheme, *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11(8), pp. 543–553, 2020.

[36] Thapar D., Jaswal G., Nigam A., Kanhangad V.: PVSNet: Palm Vein Authentication Siamese Network Trained using Triplet Loss and Adaptive Hard Mining by Learning Enforced Domain Specific Features. In: *2019 IEEE 5th International Conference on Identity, Security, and Behavior Analysis (ISBA)*, pp. 1–8, 2019.

[37] Wang P., Qin H.: Palm-vein verification based on U-Net. In: *IOP Conference Series: Materials Science and Engineering, Volume 806, International Conference on AI and Big Data Application (AIBDA 2019) 20–22 December 2019, Guangzhou, China*, vol. 806, 2019.

[38] Wen Y., Zhang K., Li Z., Qiao Y.: A Discriminative Feature Learning Approach for Deep Face Recognition. In: *Computer Vision – ECCV 2016. 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII*, pp. 499–515, Springer, 2016.

[39] Wolf L., Hassner T., Taigman Y.: Descriptor Based Methods in the Wild. In: *Workshop on Faces in "Real-Life" Images at the European Conference on Computer Vision (ECCV), Marseille, France*, pp. 1–14, 2008.

[40] Wu W., Elliott S.J., Lin S., Yuan W.: Low-cost Biometric Recognition System based on NIR Palm Vein Image, *IET Biometrics*, vol. 8(3), pp. 206–214, 2019.

[41] Zhao S., Zhang B.: Joint Constrained Least-Square Regression With Deep Convolutional Feature for Palmprint Recognition, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52(1), pp. 511–522, 2020.

[42] Zheng Q., Kumar A., Pan G.: A 3D feature descriptor recovered from a single 2D palmprint image, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38(6), pp. 1272–1279, 2016.

[43] Zhong D., Du X., Zhong K.: Decade progress of palmprint recognition: a brief survey, *Neurocomputing*, vol. 328, pp. 16–28, 2019.

Affiliations

Asma Aounallah
Echahid Cheikh Larbi Tebessi University, Laboratory of Mathematics, Informatics and Systems (LAMIS), Tebessa, 12002, Algeria, asma.aounallah@univ-tebessa.dz

Abdallah Meraoumia
Echahid Cheikh Larbi Tebessi University, Laboratory of Mathematics, Informatics and Systems (LAMIS), Tebessa, 12002, Algeria, ameraoumia@univ-tebessa.dz

Hakim Bendjenna
Echahid Cheikh Larbi Tebessi University, Laboratory of Mathematics, Informatics and Systems (LAMIS), Tebessa, 12002, Algeria, hakim.bendjenna@univ-tebessa.dz

Received: 28.12.2021
Revised: 30.05.2022
Accepted: 15.06.2022