Biometric Palmprint Verification: A Dynamical System Approach

David Palma, Pier Luca Montessoro, Giulia Giordano, and Franco Blanchini

***Abstract*—Most of the existing techniques for palmprint recognition rely on metrics, typically based on static functions, which evaluate the distance between a pair of features. In this paper, we propose a new technique for palmprint verification based on a dynamical system approach for principal palm lines matching. The proposed dynamic algorithm is recursive and involves a positive linear dynamical system, whose evolution depends on the matching level between the two input images. In a preprocessing phase, the procedure iteratively erodes both of the images to be compared, by eliminating points in each image that do not have enough close neighboring points both in the image itself and the comparison image. As a result of the iterations, only the points that have enough neighboring points in both the image itself and in the comparison image can survive. Thus, the output of the dynamical system converges either to zero, when a deep mismatch exists between the two images, or to a high value, when a good matching is observed. The results, in terms of verification, are in line with the state-of-the-art results in the current literature. The main advantage of the approach is its robustness when dealing with low-resolution and noisy images. The impact of noise (e.g., salt and pepper noise) is effectively reduced: images corrupted with such noise are easily recognized, while a randomly generated image is rejected even when compared with itself.**

***Index Terms*—Biometrics, dynamic algorithm, dynamical system, line matching, palmprint, principal lines.**

I. Introduction

With the increasing demand of automated systems based on person identification, the importance of biometric systems is growing. These systems are based on the measurable biological (anatomical and physiological) or behavioral characteristics used for the identification of an individual. Different features are used in biometric systems, such as fingerprints [20], [23], palmprint [4], [7]–[9], [12]–[14], [16], [17], [19], [21], [24], [25], [28]–[32], [35]–[42], [44], [46], [47], [52]–[63], hand geometry [26], [31], [45], [49], iris [1], [10], [51], and face [11], [18], [33], [48], [50]. Unlike conventional methods for personal identification, such as personal identification number, password, and key, these features cannot be duplicated, lost or stolen. Hence, biometric systems are suitable to be used in various fields, such as high security, forensic, and commercial applications.

Palmprint recognition, a relatively novel but promising biometric technology, has recently received considerable interest, mostly for its importance in forensics [7] (about 30% of the latents found in crime scenes are from palms [22]) and for several potential civil applications [27]. Compared with the other physical biometric characteristics, palmprint authentication has several advantages: low-resolution imaging, low-intrusiveness, stable line features, and low-cost capturing device. Indeed, since the principal lines and wrinkles can be observed under low-resolution images (e.g., 100 dpi or lower), palmprint systems do not require high resolution capturing devices [42]. However, palmprint recognition techniques based on both low- and high-resolution features are proposed in the literature (see the survey in Section II); such methods use as local features, respectively, principal lines and ridges. The approach discussed in this paper is based on low-resolution features, since it uses as local features the principal lines of the palm, which are very important physiological characteristics to distinguish between different individuals because of their stability and uniqueness. Compared with ridges and wrinkles, principal lines are usually the consequence of genetic effects: therefore, they are the most significant features in palmprint images and have good permanence. However, principal lines may be similar in different individuals, which makes their distinctiveness relatively low; for this reason, palmprint recognition is a challenging problem [60].

To address this challenge, we propose a method consisting of the following.

1) A region of interest (ROI) extraction phase, which follows the typical sequence of steps used in [35] and [60], to face different issues mainly due to nonlinear distortion, such as rotation and translation of the palm with respect to the image and nonuniform illumination.
2) An unconventional feature extraction phase based on the principal lines of the palmprint [63].
3) A novel approach to palmprint matching based on a *dynamic algorithm*. The algorithm involves a positive linear dynamical system [3], whose evolution is determined by the matching level between the two input images: its output converges to zero when the two images have a deep mismatch, while it reaches a high value in the case of good matching.

Manuscript received January 16, 2017; revised May 2, 2017; accepted October 12, 2017. Date of publication November 28, 2017; date of current version November 19, 2019. This paper was recommended by Associate Editor D. Zhang. *(Corresponding author: Pier Luca Montessoro.)*

D. Palma and P. L. Montessoro are with the Polytechnic Department of Engineering and Architecture, University of Udine, 33100 Udine, Italy (e-mail: palma.david@spes.uniud.it; montessoro@uniud.it).

G. Giordano is with the Delft Center for Systems and Control, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: g.giordano@tudelft.nl).

F. Blanchini is with the Department of Mathematics, Computer Science and Physics, University of Udine, 33100 Udine, Italy (e-mail: franco.blanchini@uniud.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2017.2771232

Methods based on dynamical systems have been successfully exploited to improve the performance of algorithms (see [2]). Here, we show that the proposed dynamic algorithm, included as a step in the matching phase, confers more robustness to palmprint recognition techniques, especially when low-resolution and noisy images are involved: the impact of noise, such as salt and pepper noise, is effectively reduced, avoiding both false rejection of images corrupted by noise and false acceptance of randomly generated images.

The results obtained by means of this novel algorithm are promising, and comparable to existing palmprint recognition algorithms. Furthermore, in order to allow implementation on mobile devices for online user authentication, each step has been specifically designed to limit memory requirement and computational complexity.

Section II of this paper provides an overview of the related work. Then, Section III presents the new approach to palmprint verification based on a noise-rejecting dynamic algorithm, while Section IV describes the main steps of the preprocessing phase and feature extraction phase that is based on the principal lines of the palmprint. Section V reports and discusses the experimental results of the developed palmprint verification system. Section VI draws conclusions.

II. Related Work

In recent years, palmprints have been investigated extensively in automated personal authentication, because of their advantages with respect to other physical biometric characteristics. Therefore, a number of interesting palmprint recognition techniques have been proposed in the literature, which can be grouped in two categories: 1) approaches based on low-resolution features and 2) approaches based on high-resolution features, which use creases and ridges, respectively, as local features. The former approach analyzes the structure of the palmprint, where generally only principal lines, wrinkles, and texture are well evident, while the latter approach tries to find the ridges or minutiae-like point features to represent the palm pattern. Palm lines include both principal lines and wrinkles, but principal lines get more attention because of their stable and highly distinctive nature [5], [34]. Hence, most authors focus on palm lines when analyzing palmprint images, and we will also focus on low-resolution methods in the following. In line-based approaches, the extracted palm lines are either matched directly or represented in other formats through some feature transformations for matching.

Zhang and Shu [62] used the datum point invariant property and the line feature matching technique to conduct the verification process via the palmprint features. This approach, in which the authors inked the palmprints on to paper and then scanned them to obtain images, is not suitable for many online security systems. Wu *et al.* [54] used the Canny operator [6] to detect palm lines and the orientations of the edge points are passed into four membership functions representing four directions. For each direction the energy is computed, then the feature value is normalized. Finally, Euclidean distance is used for matching. Wu *et al*. [57] also proposed an algorithm based on the hidden Markov models (HMMs). This approach uses Sobel masks to compute the magnitude of palm lines, then the projections along both x and y directions are used to obtain the inputs of the HMMs. A third approach proposed by Wu *et al*. [55] is based on the design of two masks which compute the vertical first- and second-order derivatives obtained by rotating the two standard masks. They identify the edge points and corresponding directions by using the zero-crossing of the first-order derivative, then the positive magnitude of the corresponding second-order derivative is considered as the magnitude of the lines. The feature vector, consisting of the weighted sum of the local directional magnitude, is normalized by its maximum and minimum elements. Finally, as in [54], Euclidean distance is used for matching. Boles and Chu [4] used Sobel masks and thresholds to construct binary edge images and then the Hough transform to detect the palm features as approximated straight lines for matching. Huang *et al.* [21] and Jia *et al.* [25] used the modified finite Radon transform (MFRAT) to extract the principal lines, then a matching algorithm based on pixel-to-area comparison measures the similarity between two palmprint images. Fei *et al.* [16] used a double half-orientation-based method for feature extraction and palmprint recognition, while in [17] they propose a double-orientation code (DOC) algorithm to represent the orientation of palmprint features, evaluating the similarity between the DOC through a nonlinear angular matching scheme. Gao and Leung [18] and Leung *et al.* [36] made use of Sobel masks to extract line-like features from the palmprints and then of line segment Hausdorff distance to compare two palmprints. Han *et al.* [19] employed Sobel and morphological operations to extract line-like features from palmprint images obtained using a scanner, and then a correlation function to measure the similarity between the two feature vectors. Similarly, for verification, Kumar *et al.* [31] used other directional masks to extract line-like features from the palmprints captured using a digital camera, then use a normalized correlation for matching. Diaz *et al.* [12] used Sobel masks and morphologic operator as two separated feature extractors to obtain the gradient of the images, which are then classified by neural networks. Kung *et al.* [33] designed a decision-based neural network classifier and applied it to face recognition and to palmprint verification. Other approaches based on low-resolution features are named subspace-based approaches, which usually adopt principal component analysis (PCA), linear discriminant analysis (LDA), and independent component analysis (ICA). Zhang *et al.* [60] used 2-D Gabor filters to extract texture features from low-resolution palmprint images captured using a CCD camera: two palmprint images are compared in terms of their Hamming distance. Connie *et al.* [8] made use of PCA and ICA to extract the palmprint texture features, then compare palmprints based on three types of classifiers: 1) Euclidean distance; 2) cosine measure; and 3) probabilistic neural network methods. Ekinci and Aykut [14] used a wavelet-based kernel PCA method by integrating the Daubechies wavelet representation of palm images and the kernel PCA method for palmprint recognition; similarity measurement is accomplished by using a weighted Euclidean linear distance-based nearest neighbor classifier.

Apart from neural networks, to the best of the authors' knowledge, dynamic algorithms have never been employed for palmprint matching. The novel method we propose is based on the evolution of a suitable dynamical system that erodes nonmatching features and enhances matching features.

III. Dynamic Algorithm for Matching

This section highlights the main idea of the proposed noise-rejecting dynamic (iterative) algorithm for palmprint matching.

Given two images (matrices of Boolean values) $X, Y \in \{0,1\}^{s\times s}$, of equal dimensions, we consider the operator

$$\left(X', Y'\right) = f(X, Y) \tag{1}$$

which provides two new images $X', Y' \in \{0,1\}^{s\times s}$, having the same size as X and Y. For our palmprint verification purposes, the operator f must be chosen to assess the "matching level" between the two initial images. To simply explain the idea, let us consider a simple (non dynamic) mismatch function. Denoting by $\Sigma(\cdot)$ the number of 1 (active) pixels in an image, we define the *matching index* between X and Y as

$$\alpha(X, Y) = \frac{\gamma(X, Y)}{2}\left[\frac{\Sigma\left(X'\right)}{\Sigma(X)} + \frac{\Sigma\left(Y'\right)}{\Sigma(Y)}\right] \tag{2}$$

where the coefficient γ is lower when the difference of active pixels in the input images is higher

$$\gamma(X, Y) = \begin{cases} 1 - \log_s\left(\dfrac{\Sigma(X)}{\Sigma(Y)}\right) & \text{if } \Sigma(X) \geq \Sigma(Y) \\ 1 + \log_s\left(\dfrac{\Sigma(X)}{\Sigma(Y)}\right) & \text{otherwise.} \end{cases} \tag{3}$$

With this choice of γ, for large differences in the number of active pixels between the two initial images, the coefficient quickly tends to a low value, thus reducing $\alpha(X, Y)$.

To assess the matching level of two images, a simple option is to consider the intersection $X' = Y' = X \cap Y$, associated with the AND logic function

$$\left[X \bigcap Y\right]_{ij} = x_{ij} \,\text{AND}\, y_{ij}.$$

However, the approach based on this choice of (1) is clearly not noise-rejecting, since it evaluates the correspondence between single pixels (isolated from their context), which may be noisy. Conversely, to avoid a misleading recognition, noisy pixels that are accidentally matching in the two images should not provide a positive contribution to the matching index.

We pursue an approach that rejects noise and, to this aim, we assume that the matching of "isolated spots" is not as significant as the matching of wide "stripes" or "islands," even if the number of matching isolated spots is very high. The operator f must then be chosen so as to set to 0 all the pixels without a sufficient number of neighboring 1 pixels, both in the image itself and in the comparison image: only significant clusters of pixels that have a corresponding cluster in the complementary image must remain active.

The desired behavior is illustrated in Fig. 1.

1) In the first example, two significant sets of active pixels are surrounded by 0 pixels in both images X and Y, which show a *good matching*. Actually, almost all of the active pixels should survive in the transformed images $Z = X'$ and $W = Y'$.
2) In the second example, there is a just *partial superposition*. Hence, in both of the transformed images, the sets of active pixels should be partially eroded.
3) In the third example, there is a *perfect superposition* between *isolated points*. In this case, the transformed images should be almost empty.

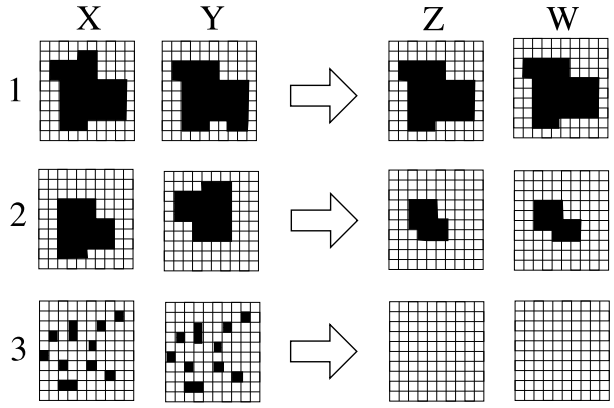


Fig. 1. Desired matching patterns: 1 good matching; 2 partial matching; and 3 no matching. X and Y are the initial images, $Z = X'$ and $W = Y'$ the desired transformed images.

As is experimentally shown later, the behavior in Fig. 1 can be achieved with the proposed dynamic (iterative) algorithm.

Remark 1: In view of the iterative nature of the algorithm, we only need to emphasize the presence of neighborhoods in the comparison image, since in at most two steps this indirectly emphasizes neighborhoods in the image itself.

We assume that the images A and B, provided as inputs, have already undergone a preliminary processing (thoroughly described in Section IV). Given the set

$$\mathcal{N}_{ij} = \{h, l : |h - i| \leq \nu, \quad |l - j| \leq \nu, \quad h, l \in \mathbb{Z}\}$$

of neighboring points of (i, j) within a "radius" of integer amplitude $\nu > 0$, the "fading factor" $0 < \lambda < 1$, the coefficient $\mu > 0$ that emphasizes the presence of neighbors in the comparison image and the tolerance $\epsilon > 0$, Algorithm 1 works as follows.

Note that the images are processed as real-valued matrices, in view of step 1. Due to the componentwise product at step 4, all the pixels that are initially 0 remain 0: the procedure can "inactivate" active pixels, but cannot "activate" inactive pixels. As a consequence, the index α is always a number $0 \leq \alpha \leq 1$. The rationale of step 3 (the core of the procedure) is the following. Assume that a pixel has no match on the other image. Then the corresponding equation becomes

$$x_{ij}(k+1) = \lambda x_{ij}(k).$$

Hence, being $\lambda < 1$, the value of pixels having no match on the other image quickly converges to zero: typically, in few iterations $x_{ij}(k)$ goes below the threshold ϵ. We stress that, if a pixel has no match at the beginning, this status is preserved for all the steps of the procedure, due to step 4.

Conversely, the value of pixels having a large neighborhood in the comparison image diverges. If the same condition

Algorithm 1 Matching Index Computation

Input: Boolean images A and B.

Parameters: Number of steps N, positive constants $\lambda < 1$ and μ, integer neighborhood amplitude $\nu > 0$, a small tolerance $\epsilon > 0$.

Outputs: Matching index $\alpha(A, B)$.

1) Convert the two input images from boolean into real matrices $X \coloneqq A$ and $Y \coloneqq B$.
2) Set $k = 0$.
3) At each iteration, compute the updated values for each pixel in both images

$$x_{ij}(k+1) = \lambda x_{ij}(k) + \mu \sum_{hl \in \mathcal{N}_{ij}} y_{hl}(k) \tag{4}$$

$$y_{ij}(k+1) = \lambda y_{ij}(k) + \mu \sum_{hl \in \mathcal{N}_{ij}} x_{hl}(k) \tag{5}$$

4) At each iteration, set $X \coloneqq X \times A$, $Y \coloneqq Y \times B$, where $\times$ denotes the componentwise product (i.e., $x_{ij} \coloneqq x_{ij} \times a_{ij}$ for all i, j).
5) Set $k = k + 1$ and, IF $k < N$, GOTO step 3.
6) Convert the matrices to boolean, $[A', B'] = \text{logic}(X, Y)$, as follows: IF $x_{ij} < \epsilon$, THEN $a'_{ij} \coloneqq 0$; ELSE $a'_{ij} \coloneqq 1$ (resp. IF $y_{ij} < \epsilon$, THEN $b'_{ij} \coloneqq 0$; ELSE $b'_{ij} \coloneqq 1$).
7) Compute the matching index α as in (2).

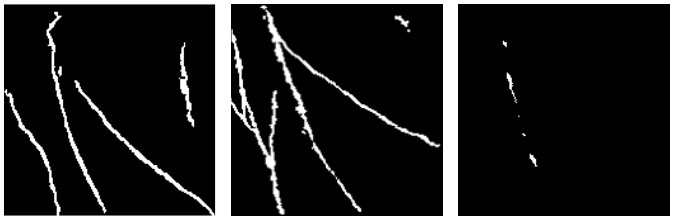


Fig. 2. Dynamic algorithm behavior. (a) and (b) Similar principal lines extracted from two different subjects, provided as input, and (c) image resulting from the application of the dynamic algorithm.

is true in the complementary neighborhood, then we are in the presence of two matching groups that "cooperate" by increasing each other. In this case, the values of x_{ij} and y_{ij} significantly increase in few iterations. The exponential convergence/divergence of the procedure assures that, in few iterations, some pixels become practically zero (eventually set to 0), and others achieve large values (eventually set to 1).

To exemplify the algorithm behavior, suppose that the two input images A and B are those shown in Fig. 2(a) and (b). Then, the result of the dynamic algorithm, $A' = B'$, is shown in Fig. 2(c). In both images, the algorithm has iteratively eliminated points that do not have enough close neighboring points in the comparison image; the number of survived pixels is very limited, even when starting from images having similar principal lines.

A considerable advantage of our dynamic approach is its robustness with respect to noise. Images corrupted, e.g., with salt and pepper noise are easily recognized, while an image randomly generated is rejected even if compared with itself.

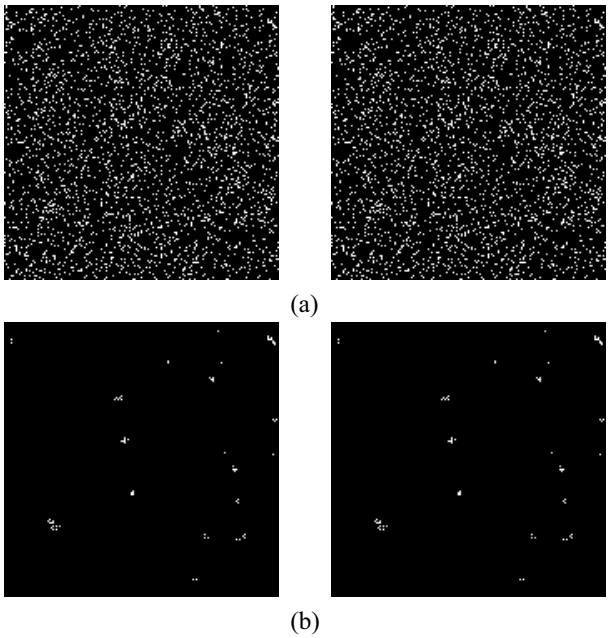


Fig. 3. Salt-and-pepper noise. (a) A and B ($A = B$). (b) A' and B' ($A' = B'$).

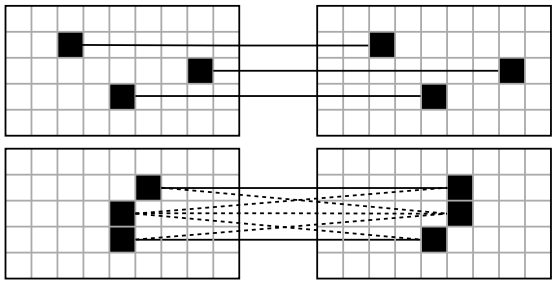


Fig. 4. Dynamic algorithm example: (top) pairs of isolated points and (bottom) clusters of points. Points in the same positions are connected through solid lines, whereas points in different positions but close enough (the radius is $\nu = 1$) are connected through dashed lines.

Example 1 (Salt-and-Pepper Noise): To explain a peculiar feature of the algorithm, we compare a salt-and-pepper noisy image with itself (see Fig. 3). The algorithm eliminates almost all of the pixels and the resulting *matching index* is extremely low.

The idea of the procedure can be described as follows. Consider a family of bipartite graphs, whose nodes are the active pixels in the two images (A and B) to be compared. An arc connects each pair of active pixels that occupy neighboring positions in the two images. In this way, we form clusters of active pixels. A cluster is preserved only if it includes enough pixels, otherwise it fades away. In the case of perfect matching of two images with isolated points (as in Fig. 4, top), there are just isolated pairs of nodes, which disappear when the dynamic algorithm evolves[1] (local stability and convergence to 0). Conversely, in the presence of partially matching blobs

[1]Although the term "to evolve" is used, the algorithm is not based on machine-learning models. We want to emphasize the iterative nature of the dynamic algorithm that converges to the final state by eroding the pixels that do not match within a neighborhood.

(as in Fig. 4, bottom), the clusters of nodes are numerous enough to survive (local instability with saturation to 1).

A. Properties of the Algorithm and Parameter Tuning

We analyze here the algorithm based on the linear recursive equations (4) and (5), to better understand its behavior.

First of all, the parameters λ and μ in the procedure need to be tuned. We remind that:

1) the fading parameter $\lambda < 1$ cancels "isolated pixels";
2) the emphasis parameter μ increases "surrounded pixels."

Consider a generic neighborhood $\mathcal{N}$ of size ν and assume that all the points in $\mathcal{N}$ are initially active (set to 1) in both A and B. Then, their values should "blow up." Given the updating equation for x_{ij}

$$x_{ij}(k+1) = \lambda x_{ij}(k) + \mu \sum_{hl \in \mathcal{N}_{ij}} y_{hl}(k)$$

let m be the number of pixels in the considered neighborhood. Being all the pixels initially set to 1, a blow-up condition is

$$\lambda + \mu m > 1 \tag{6}$$

(the lower bound in Fig. 9). Therefore, at step $k = 1$, all the surrounded pixels reach a value greater than 1; at step $k = 2$, the value becomes even bigger; an so on.

Let us now consider a partial superposition, in which only a percentage $p < 1$ of the pixels in the neighborhood are 1. We may take p as a threshold fraction, so that 1-pixels with less than a fraction p of 1-pixels in the complementary neighborhood tend to fade. This leads to the criterion

$$\lambda + p\mu m < 1 \tag{7}$$

(the upper bound in Fig. 9). Therefore, given the neighborhood radius ν and the number of iterations N, a criterion to choose the parameters λ and μ is the following.

1) Fix λ based on the desired fading speed.
2) Compute m, the neighborhood size, in terms of pixels.
3) Fix p, the desired threshold fraction.
4) Compute μ which satisfies (6) and (7).

Property 1 (Decoupling): Different regions that are not connected in either A or B do not interact and have separate evolutions, as illustrated in Fig. 5. Hence, isolated spots can be analyzed separately.

Property 2 (Convergence/Divergence): Two overlapping spots in A (containing q pixels) and in B (containing s pixels) give rise to a positive linear system, as described next. Let us group in a vector $x(k) \in \mathbb{R}^q$ the variables x_{ij} of the spot in A and in a vector $y(k) \in \mathbb{R}^s$ the variables y_{ij} of the spot in B.

The system evolves as follows:

$$\begin{bmatrix} x(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} \lambda I & \mu\Omega \\ \mu\Omega^\top & \lambda I \end{bmatrix} \begin{bmatrix} x(k) \\ y(k) \end{bmatrix} \tag{8}$$

where I is the identity matrix (of the proper dimension), λ and μ are the previously fixed constants and Ω is a connection matrix defined as

$$\Omega_{ij} = \begin{cases} 1, & \text{if pixel } y_j \text{ is in the neighborhood of pixel } x_i \\ 0, & \text{otherwise.} \end{cases}$$

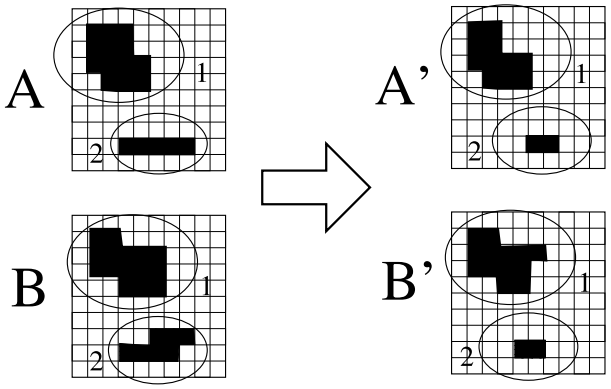


Fig. 5. Two sets of pixels with separate evolutions.

Note that the matrix

$$M = \begin{bmatrix} \lambda I & \mu\Omega \\ \mu\Omega^\top & \lambda I \end{bmatrix}$$

appearing in (8) is symmetric. In fact, if pixel y_j is in the neighborhood of pixel x_i, also x_i is in the neighborhood of y_j.

Then, the fading or the permanence of the two connected spots is doomed by the convergence or divergence of the linear system (8). The linear system converges (i.e., is stable) if and only if the dominant eigenvalue $\sigma_{\max}$ (i.e., the eigenvalue with maximum modulus) of matrix M is smaller than 1. Conversely, the linear system diverges (i.e., is unstable) if and only if $\sigma_{\max}$ is greater than 1. Since matrix M is non-negative, its dominant eigenvalue $\sigma_{\max}$ is real and positive. Moreover, $\sigma_{\max}$ is an increasing function of both λ and μ, and increases if the number of 1 entries in matrix Ω increases.

This explains why the pair of identical images with salt and pepper noise leads to a poor matching index: even if there is actually a good (or perfect) matching between the two images, there are not enough connections between the active pixels of one image and the active pixels of the other.

Since in the average, for the salt and pepper noise, half of the pixels in the complementary neighborhood are inactive, we recommend to choose

$$p \approx 1/2.$$

Then, in view of (6) and (7), this leads to the conditions

$$\frac{1}{2} < \frac{1-\lambda}{\mu m} < 1.$$

IV. Palmprint Image Processing

Fig. 6 shows the overall block diagram of palmprint image processing that consists of two main phases: 1) preprocessing and 2) feature extraction.

A. Preprocessing

Before the feature extraction step, a preprocessing is essential in order to obtain a subimage denominated ROI from the captured palmprint image. In fact, usually palmprint images can have different orientation and size, and are also subject to noise. Moreover, the region of not-interest (e.g., fingers,

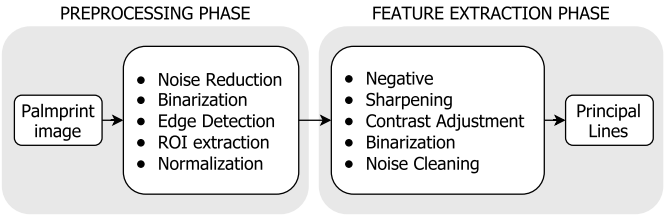


Fig. 6. Block diagram of palmprint image processing.

wrist, and image background) may affect the accuracy in processing and verification performance. First of all, noise is reduced by proper filters, then the ROI is extracted. In order to align the different palmprint images, 1) binarization and 2) edge detection are executed. After alignment 3) ROI is extracted and 4) normalized to have a specific mean and variance for all images [20]. These four steps are shown in Fig. 7 and described more in detail below.

1) Binarization: Binarization converts a gray level image $I(x, y)$ into a binary image $B(x, y)$, where the hand is the foreground and the rest is background. Thus, a local adaptive thresholding can be applied to binarize the image.

After partitioning the image into nonoverlapping sub-blocks of size $N \times N$, for each sub-block we compute a threshold value by using Otsu's method [43]: the image pixels are split into two classes (background and object), and the best threshold value t is obtained based on the maximum value of the variance between the two classes [15]. Then, for each sub-block, we apply a global thresholding as follows:

$$B(x, y) = \begin{cases} 1 & \text{if } I(x, y) \geq t \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

The image after the binarization phase is shown in Fig. 7(b).

2) Edge Detection: To detect the hand edge, the segmented binary image is filtered by using the Canny operator [6]. This edge detector ensures good noise immunity and at the same time detects true edge points with minimum error [15]. A practical approximation of the Canny filter is the following.

1) Convolve the input image $I(x, y)$ with a 2-D Gaussian filter $G(x, y)$

$$H(x, y) = I * G(x, y). \tag{10}$$

In order to reduce the computational cost, a convolution with a 2-D Gaussian can be separated into two convolutions with 1-D Gaussians.

2) Apply the gradient operator and calculate the gradient magnitude M and orientation Θ as follows:

$$H_x(x, y) = \frac{\partial H(x, y)}{\partial x} \qquad H_y(x, y) = \frac{\partial H(x, y)}{\partial y} \tag{11}$$

$$M(x, y) = \sqrt{H_x^2(x, y) + H_y^2(x, y)} \tag{12}$$

$$\Theta(x, y) = \arctan\left(\frac{M_y(x, y)}{M_x(x, y)}\right). \tag{13}$$

3) Apply the nonmaximum suppression algorithm that removes the nonmaximum pixels, while preserving the connectivity of the contours, as follows: for each pixel (x, y), if one of the neighboring pixels in the orthogonal directions to edge orientation is greater than the current pixel, then discard it.

4) The last step is the hysteresis thresholding, which uses two thresholds: T_H, calculated by using Otsu's algorithm [15], and $T_L = (1/2)T_H$. This algorithm sets a pixel (x, y) to 0 if it has a value less than T_L, and to 1 if it has a value greater than T_H or it is connected to a pixel that has a value greater than T_H.

The image after the edge detection phase is in Fig. 7(c).

3) Region of Interest Extraction: In this last step, the palmprint image is aligned in a standard pose, to reduce issues due to nonlinear factors, such as nonuniform illumination, rotation, and translation. In order to extract the central part of the palmprint images, we detect two reference points between the fingers, as shown in Fig. 7(d). These points are used to construct a reference line for aligning the different palmprint images, and the middle point between them is used to determine the central position of the ROI. Finally, the ROI can be extracted as shown in Fig. 7(e).

4) Normalization: In order to have a prespecified mean and variance for all palmprint images, the extracted ROI images are normalized to reduce the possible nonuniform illumination and noise, as suggested in [23]. Therefore, given the input ROI image $I(x, y)$, the normalized image is defined as follows:

$$I'(x, y) = \begin{cases} \mu_n + \rho & \text{if } I(x, y) > \sigma^2 \\ \mu_n - \rho & \text{if } I(x, y) \leq \sigma^2 \end{cases} \tag{14}$$

where

$$\rho = \sqrt{\frac{\sigma_n^2 (I(x, y) - \mu)^2}{\sigma^2}}. \tag{15}$$

The parameter values μ_n and σ_n^2 used for the computation are the same suggested in [23], thus we set both the mean and variance values to 100. The result is shown in Fig. 7(f).

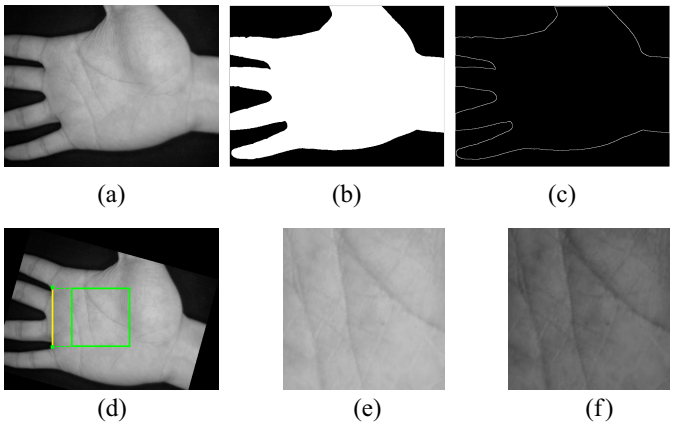


Fig. 7. Main steps of preprocessing. (a) Original palmprint image. (b) Noise reduction and binarization by using local adaptive thresholding. (c) Edge detection. (d) Scaling, alignment and ROI location. (e) ROI extraction. (f) Normalization.

B. Feature Extraction

Once the central part (ROI) is segmented, features can be extracted for matching. The feature extraction phase plays an

important role in image identification and verification. There are many features exhibited in a palm, such as the three principal lines caused by flexing hand and wrist in the palm, which are denominated as heart line, head line, and life line, respectively. We use a principal line extraction method based on [63], whose main steps are illustrated in Fig. 6: 1) conversion to a negative image; 2) sharpening of the image with Top-Hat filter and gradient magnitude computation; 3) linear contrast adjustment; 4) binarization with Otsu's method; and 5) noise cleaning with median filter.

1) Negative: After normalization in the last step of the preprocessing phase, the resulting enhanced ROI image $I(x, y)$ is converted to its negative as follows:

$$I'(x, y) = \max\{I(x, y)\} - I(x, y). \tag{16}$$

In this way, the principal lines of the palm have high intensity.

2) Sharpening: To correct uneven illumination we use a Top-Hat transform, then apply a linear contrast enhancement to the output image. The Top-Hat transform enables us to extract small bright objects from a varying background and is defined as the difference between the input image and its morphological opening by using a defined structuring element ρ. The opening of $I(x, y)$ by ρ is obtained by the erosion of $I(x, y)$ by ρ, followed by dilation of the resulting image by ρ:

$$I \circ \rho = (I \ominus \rho) \oplus \rho = \bigcup_{I(x,y) \subset \rho} I(x, y). \tag{17}$$

Hence, the opening of $I(x, y)$ by ρ is the union of all sets $I(x, y)$ contained in ρ. Then, the opening Top-Hat transformation of image I, denoted by TH, is given by

$$\mathrm{TH} = I - (I \circ \rho) \tag{18}$$

that represents the difference set of the domain of $I(x, y)$ and the opening domain of $I(x, y)$. After the morphological operations, the gradient magnitude is computed as described in the preprocessing phase: this sharpening operation returns the bright spots of the image that are smaller than the structuring element.

3) Contrast Adjustment: In order to improve the contrast of the image, a linear contrast enhancement is applied by stretching the range of intensity values it contains to a specific range of values. Before applying the linear scaling function it is necessary to identify the lower and upper bounds from the histogram, which are the minimum and maximum brightness values in the image, and to apply a transformation to stretch this range to fill the full range $(0, 255)$. Let min and max be the lower and the upper limits of the histogram, respectively, then each pixel of the image is scaled to the range $(0, 255)$ through the following equation:

$$I'(x, y) = (I(x, y) - \min)\left(\frac{255}{\max - \min}\right). \tag{19}$$

4) Binarization: A global thresholding is applied at the gray level image resulting from the previous sharpening filter and linear contrast adjustment. The threshold is calculated with Otsu's method [43] described in the preprocessing phase.

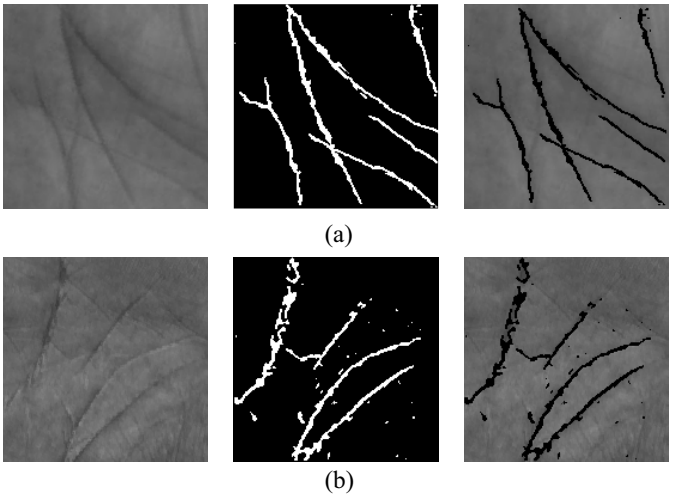


Fig. 8. Some results of palm line extraction. (a) Feature extraction on CASIA database. (b) Feature extraction on PolyU database (lower resolution).

5) Noise Cleaning: As a last step in the feature extraction phase, a median filter is used in order to remove noise and trivial lines from the image. Fig. 8 shows examples of extracted feature with the algorithm described above.

V. Experimental Results

A. Databases Used in Simulation

The performance of the proposed palmprint verification system has been tested upon two databases that have been worldwide shared for research purposes: 1) *CASIA palmprint database* [65] and 2) *PolyU palmprint database II* [64].

The first database contains 5502 palmprint images with 8 bit gray-level of size 640×480 pixels at 96 dpi resolution captured from 312 subjects by a low resolution CCD-based device. For each subject there are palmprint images from both left and right hand captured eight times at different times from people of different ages. The second database, which has a lower resolution, contains 7752 palmprint images with 8 bit gray-level of size 384×284 pixels at 75 dpi resolution captured from 386 palms of 193 subjects: these palmprint images have been collected in two sessions with the average interval over two months, where about ten samples have been captured from each palm for each session.

We have performed experiments by taking: 1) 16 sample palmprint images of the left and right hands (eight images for hand) of all 312 subjects, for a total of 4992 images, from the *CASIA palmprint database* and 2) ten sample palmprint images of 381 individuals related to the first and second sessions, for a total of 7620 images, from the *PolyU palmprint database II*. This setup constitutes a total number of experiments equal to: 1) $\binom{4\,992}{2} = 12\,457\,536$, including $312 \times \binom{16}{2} = 37\,440$ genuine experiments (the others are impostor experiments), for the *CASIA palmprint database* and 2) $\binom{7\,620}{2} = 29\,028\,390$, including $381 \times \binom{20}{2} = 72\,390$ genuine experiments, for the *PolyU palmprint database*.

B. Criteria for Performance Evaluation

To evaluate the verification accuracy of the proposed method, each palmprint image in the database has been matched against the other palmprint images. The matching between palmprints captured from the same palm is defined as a genuine matching or intraclass matching, otherwise as an impostor matching or interclass matching.

The general method to evaluate the performance of a palmprint authentication system is based on the false acceptance rate (FAR) and the false rejection rate (FRR) [41]. The FAR is defined as the percentage of invalid inputs which are incorrectly accepted and is computed as the number of accepted imposter claims over the total number of imposter accesses. The FRR is defined as the percentage of valid inputs which are incorrectly rejected and is computed as the number of rejected genuine claims over the total number of genuine accesses. FAR and FRR are closely related because the increase of the one implies the decrease of the other. Mathematically these two indices are expressed as follows:

$$\mathrm{FAR} = \frac{1}{N}\sum_{k=1}^{N}\mathrm{FAR}(k) \tag{20}$$

$$\mathrm{FRR} = \frac{1}{N}\sum_{k=1}^{N}\mathrm{FRR}(k) \tag{21}$$

where

$$\mathrm{FAR}(k) = \frac{\text{number of FARs}}{\text{number of impostor accesses}} \tag{22}$$

$$\mathrm{FRR}(k) = \frac{\text{number of FRRs}}{\text{number of genuine accesses}}. \tag{23}$$

Other evaluation parameters are as follows.

1) *Receiver Operating Characteristic (ROC):* A graphical plot that illustrates the tradeoff between the FAR and the FRR.
2) *Equal Error Rate (EER):* The error rate at the specific threshold s for which both acceptance and rejection errors are equal, which can be easily obtained from the ROC curve.
3) *Genuine Acceptance Rate (GAR):*

$$\mathrm{GAR} = 1 - \mathrm{FRR}. \tag{24}$$

4) *GAR at a Specific FAR (GAR_x):*

$$\mathrm{GAR}_x = \mathrm{GAR}\big|_{\mathrm{FAR}=10^{-x}} \tag{25}$$

for example, considered an $\mathrm{FAR} = 0.01$, then $\mathrm{GAR}_2 = \mathrm{GAR}\big|_{\mathrm{FAR}}$.

C. Parameter Tuning for Performance Evaluation

Since the verification rate of the system varies depending on the values of λ and μ in the dynamic algorithm, these parameters need to be properly set, as discussed in Section III. Thus, given the set $\mathcal{N}_{i,j}$ of neighboring points of the generic point $p(i, j)$ we can choose the fading factor $0 < \lambda < 1$ and the coefficient $\mu > 0$. Fig. 9 depicts the region of convergence of the system according to (6) and (7), which represent the upper and lower bounds of the region, respectively. Since the thickness of the palm lines consists of a few pixels, it is reasonable to consider a square matrix with radius $\nu = 2$ (which amounts to considering a number of neighboring pixels equal to $m = 25$) for a perfect coverage of a palm line avoiding excessive overlaps with other palm lines—or background—contained in the image to be compared.

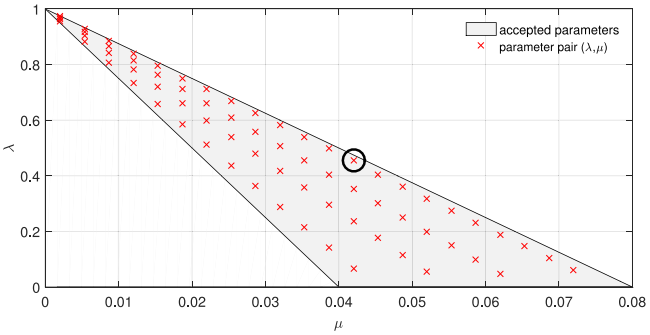


Fig. 9. Region of convergence of the system delimited by 6 and 7 [with $m = 25$ and $p = (1/2)$], and values of the parameter pairs (λ, μ). Best results have been obtained by the highlighted pair of values.

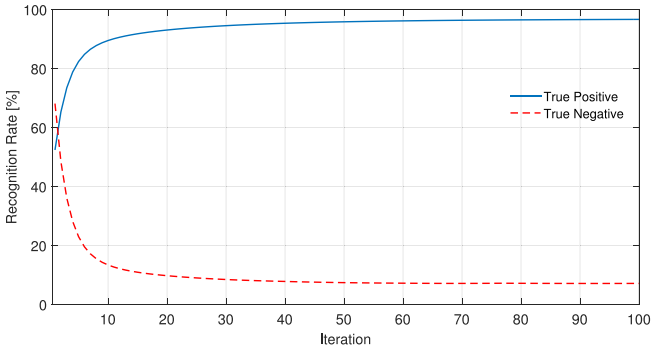


Fig. 10. Matching value against number of iterations for two palmprint features in the case of true positive recognition and true negative recognition.

Therefore, to determine the value of the parameters that guarantees the best performance, a test over a subset of the CASIA palmprint database for many different values of the parameter pairs has been performed. In this phase, we have verified every test pair for each of the palmprint images in the database and for each pair of the parameter values.

The subset of the CASIA palmprint database consists of 360 palmprint images and the number of parameter pairs (λ, μ) inside the region of convergence is equal to 70, thus the amount of the tests performed is 4 523 400. As a result of these tests we found that $\lambda = 0.465$ and $\mu = 0.042$ provide the best results. These values have been used in the experiments for both the databases in order to test the performance of the proposed method, even without parameter retuning for the PolyU database.

It is also important to choose a fixed number of iterations, after which we can rely on convergence of the dynamic system.

From Fig. 10 it is clear that, after few iterations, the reached matching value can be considered good enough with respect to the highest value achieved. Thus, to save computation time, the number of iterations for the tests has been fixed to 20.

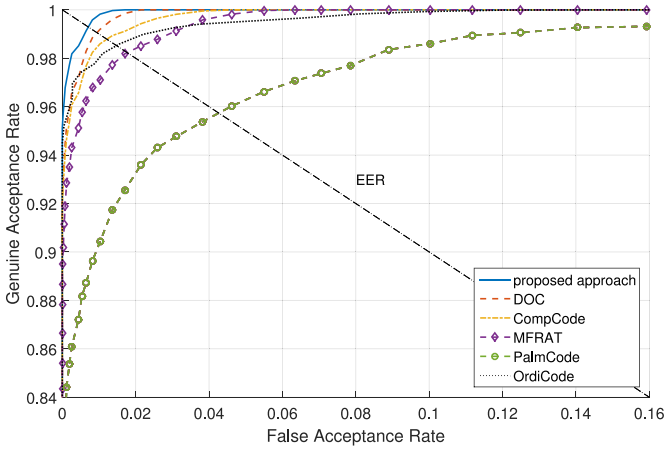


Fig. 11. Comparative GAR against FAR graphs, for the proposed approach and for other methods on PolyU palmprint database.

TABLE I
COMPARISON OF VERIFICATION RESULTS FOR DIFFERENT METHODS [16], [40]

Experimental Results (EER)

Method	CASIA	PolyU
CompCode	0.0201	0.0122
PalmCode	0.0367	0.0432
OrdiCode	0.0175	0.0150
MFRAT	–	0.0180
DOC	–	0.0092
Proposed method	0.0123	0.0071

D. Performance Comparison

The performance results of the developed palmprint recognition system are here compared with other algorithms by using the datasets as previously described.

For comparison purposes, some representative techniques in palmprint recognition have been used.

1) CompCode [29], which uses six real parts of the Gabor filters to obtain orientation information from palm lines and subsequently applies Hamming distance for feature matching.
2) PalmCode [60], which uses 2-D Gabor phase encoding scheme to extract the feature, then Hamming distance is used for matching.
3) OrdiCode [52], that compares two line-like image regions, which are orthogonal in orientation and it generates one bit feature code, then for matching is also used the Hamming distance.
4) Huang *et al*. algorithm [21], which uses the MFRAT to extract palmprint principal lines, then a matching algorithm based on pixel-to-area comparison has been designed to measure the similarity.
5) DOC [17] which stands for double-orientation code used to represent the orientation feature of palmprint which are compared with a nonlinear angular matching score method.

Table I lists the verification results, expressed in terms of EER, for different approaches, on databases that contain palmprints captured through touch-based (PolyU) and touchless (CASIA) devices. From the results, it is clear that the CASIA database is more challenging than the PolyU database in terms of palmprint verification. The proposed method achieves good results, outperforming some of the other approaches.

To better illustrate the verification performances, the ROC curves of the previous five methods and the proposed algorithm are illustrated in Fig. 11: the ROC curves are obtained by plotting GAR against FAR. The dynamic algorithm obtains a GAR_2 that is almost 100% and a GAR_3 that is greater than 97%.

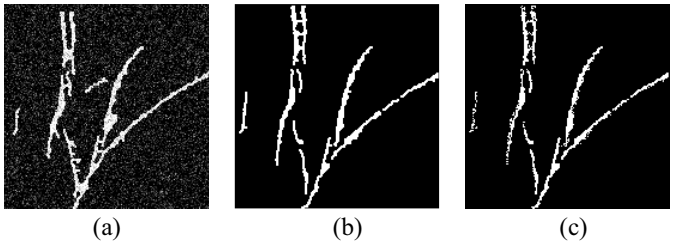


Fig. 12. Dynamic algorithm behavior on different principal line images from the same subject. (a) First image affected by white Gaussian noise with $\mu = 0$ and $\sigma^2 = 0.05$, (b) second image to be compared, and (c) surviving points after the algorithm evolution.

TABLE II
ROC COMPARISON ON BOTH THE DATABASES WITH NORMAL PALMPRINTS AND PALMPRINTS CORRUPTED BY ADDITIVE WHITE GAUSSIAN NOISE

Noise Rejection Experiments

	GAR [%] at specific value of FAR [%]		
Database	FAR= 10^{-3}	FAR= 10^{-2}	FAR= 10^{-1}
CASIA	96.85	98.63	100
CASIA + AWGN	91.07	94.75	98.63
PolyU	97.11	99.82	100
PolyU + AWGN	91.13	96.25	100

E. Robustness to Noise

To conduct experiments on noisy palmprints and demonstrate the robustness of the dynamic algorithm with respect to noise, the system has been tested by comparing palmprints corrupted by white Gaussian noise against normal palmprints. The experiments are carried out by corrupting the images with a white Gaussian noise of mean $\mu = 0$ and variance $\sigma^2 = 0.05$. Fig. 12 depicts an example of the dynamic algorithm behavior: it can be seen from the output image that the subject is easily recognized, validating the noise rejection property described in Section III.

Table II reports the result of noise rejection experiments and shows that the performance of the system is not that much degraded: even in presence of noise, the GAR_1 is greater than 98.6%. However, we have obtained better results on the PolyU database achieving a GAR of 97.8% with an EER equal to 0.022 against a GAR of 96.9% with an EER equal to 0.031 on the CASIA database. Therefore, the experiments show that the

proposed algorithm can recognize palmprints even in presence of a reasonable amount of noise.

F. Computation Time

The experiments for the proposed approach have been conducted on a virtualized machine equipped with two dedicated processors and 2048 MB RAM hosted on an Intel Core i7-4510U CPU (2.6 GHz), 8 GB RAM running a 32-bit Microsoft Windows 10. The code has been implemented using MATLAB 8.0 and to get the following results each part of code has been performed for 200 times, then the average time has been considered. Thus, the computational times required for preprocessing, palm line extraction, and palmprint matching are 376, 49, and 295 ms, respectively, thus the average response time for verification is about 0.72 s, which is more suitable for a real-time biometric verification system rather than a real-time biometric identification system. However, it is possible to further reduce the computation time, since the program codes have not completely optimized.

VI. Conclusion

In this paper, a novel approach has been presented to authenticate individuals based on their palmprint features. As a main contribution, a new recursive, dynamic algorithm has been applied for the matching of features. A noticeable advantage of such an approach is its robustness with respect to noise: for instance, images corrupted with salt and pepper noise are easily recognized, whereas an image randomly generated is rejected even when compared with itself [44]. The images provided as an input to the dynamic algorithm have undergone an image processing based on two phases: the first involving preprocessing operations to make the system invariant to rotation and translation of the palm with respect to the image and the second consisting of a sequence of robust feature extraction steps that allow to detect the principal lines of the palm without introducing noise. Extensive experiments have been conducted to evaluate the performance of the system and the results obtained from the tests clearly demonstrate the effectiveness of the proposed technique, which is able to produce promising result. As a matter of fact, the experimental results presented in Section V show that the level of GAR_1 obtained is 100% on both the databases, whereas in the noisy test GAR_1 can still be considered greater than 98.6% for the CASIA database and equal to 100% for the PolyU database. Thus, the system is comparable with existing biometric recognition systems based on palmprint recognition and other hand-based biometric technologies, including hand geometry and fingerprint verification. In particular, the achieved level of GAR is compliant with the strict requirements of high security applications.

References

[1] S. Alkassar, W. L. Woo, S. S. Dlay, and J. A. Chambers, "Robust sclera recognition system with novel sclera segmentation and validation techniques," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 3, pp. 474–486, Mar. 2016.

[2] F. Blanchini, D. de Caneva, P. L. Montessoro, and D. Pierattoni, "Control-based p-persistent adaptive communication protocol," *ACM Trans. Auton. Adapt. Syst.*, vol. 7, no. 2, pp. 1–18, 2012.

[3] F. Blanchini and S. Miani, *Set-Theoretic Methods in Control* (Systems & Control: Foundations & Applications). Boston, MA, USA: Birkhäuser, 2008.

[4] W. W. Boles and S. Y. T. Chu, "Personal identification using images of the human palm," in *Proc. IEEE Reg. 10 Annu. Conf. Speech Image Technol. Comput. Telecommun.*, vol. 1. Brisbane, QLD, Australia, 1997, pp. 295–298.

[5] A. Bruno, P. Carminetti, V. Gentile, M. La Cascia, and E. Mancino, "Palmprint principal lines extraction," in *Proc. IEEE Biometric Meas. Syst. Security Med. Appl.*, Rome, Italy, 2014, pp. 50–56.

[6] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[7] R. Cappelli, M. Ferrara, and D. Maio, "A fast and accurate palmprint recognition system based on minutiae," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 956–962, Jun. 2012.

[8] T. Connie, A. Teoh, M. Goh, and D. Ngo, "Palmprint recognition with PCA and ICA," in *Proc. Image Vis. Comput.*, 2003, pp. 227–232.

[9] T. Connie, A. T. B. Jin, M. G. K. Ong, and D. N. C. Ling, "An automated palmprint recognition system," *Image Vis. Comput.*, vol. 23, no. 5, pp. 501–515, 2005.

[10] J. Daugman, "The importance of being random: Statistical principles of iris recognition," *Pattern Recognit.*, vol. 36, no. 2, pp. 279–291, 2003.

[11] M. De Marsico, M. Nappi, D. Riccio, and H. Wechsler, "Robust face recognition for uncontrolled pose and illumination changes," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 1, pp. 149–163, Jan. 2013.

[12] M. R. Diaz, C. M. Travieso, J. B. Alonso, and M. A. Ferrer, "Biometric system based in the feature of hand palm," in *Proc. 38th Annu. Int. Carnahan Conf. Security Technol.*, Albuquerque, NM, USA, 2004, pp. 136–139.

[13] N. Duta, A. K. Jain, and K. V. Mardia, "Matching of palmprints," *Pattern Recognit.*, vol. 23, no. 4, pp. 477–485, 2002.

[14] M. Ekinci and M. Aykut, "Palmprint recognition by applying wavelet-based kernel PCA," *J. Comput. Sci. Technol.*, vol. 23, no. 5, pp. 851–861, 2008.

[15] M. Fang, G. Yue, and Q. Yu, "The study on an application of Otsu method in canny operator," in *Proc. Int. Symp. Inf. Process. (ISIP)*, 2009, pp. 109–112.

[16] L. Fei, Y. Xu, and D. Zhang, "Half-orientation extraction of palmprint features," *Pattern Recognit. Lett.*, vol. 69, pp. 35–41, Jan. 2016.

[17] L. Fei, Y. Xu, W. Tang, and D. Zhang, "Double-orientation code and nonlinear matching scheme for palmprint recognition," *Pattern Recognit.*, vol. 49, pp. 89–101, Jan. 2016.

[18] Y. Gao and M. K. H. Leung, "Face recognition using line edge map," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 764–779, Jun. 2002.

[19] C.-C. Han, H.-L. Cheng, C.-L. Lin, and K.-C. Fan, "Personal authentication using palm-print features," *Pattern Recognit.*, vol. 36, no. 2, pp. 371–381, 2003.

[20] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 777–789, Aug. 1998.

[21] D.-S. Huang, W. Jia, and D. Zhang, "Palmprint verification based on principal lines," *Pattern Recognit.*, vol. 41, no. 4, pp. 1316–1328, 2008.

[22] A. K. Jain and J. Feng, "Latent palmprint matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1032–1047, Jun. 2009.

[23] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Trans. Image Process.*, vol. 9, no. 5, pp. 846–859, May 2000.

[24] W. Jia, R.-X. Hu, Y.-K. Lei, Y. Zhao, and J. Gui, "Histogram of oriented lines for palmprint recognition," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 3, pp. 385–395, Mar. 2014.

[25] W. Jia, D.-S. Huang, and D. Zhang, "Palmprint verification based on robust line orientation code," *Pattern Recognit.*, vol. 41, no. 5, pp. 1504–1513, 2008.

[26] W. Kang and Q. Wu, "Pose-invariant hand shape recognition based on finger geometry," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 11, pp. 1510–1521, Nov. 2014.

[27] A. W. K. Kong, D. Zhang, and M. Kamel, "A survey of palmprint recognition," *Pattern Recognit.*, vol. 42, no. 7, pp. 1408–1418, 2009.

[28] A. W. K. Kong, D. Zhang, and W. Li, "Palmprint feature extraction using 2-D Gabor filters," *Pattern Recognit.*, vol. 36, no. 10, pp. 2339–2347, 2003.

[29] A. W.-K. Kong and D. Zhang, "Competitive coding scheme for palmprint verification," in *Proc. 17th Int. Conf. Pattern Recognit. (ICPR)*, vol. 1. Cambridge, U.K., 2004, pp. 520–523.

[30] K. Krishneswari and S. Arumugam, "Intramodal feature fusion based on PSO for palmprint authentication," *J. Image Video Process.*, vol. 2, no. 4, pp. 435–440, 2012.

[31] A. Kumar, D. C. M. Wong, H. C. Shen, and A. K. Jain, *Personal Verification Using Palmprint and Hand Geometry Biometric* (Lecture Notes in Computer Science). vol. 2688. Berlin, Germany: Springer, 2003, pp. 668–678.

[32] A. Kumar and D. Zhang, "Personal authentication using multiple palmprint representation," *Pattern Recognit.*, vol. 38, no. 10, pp. 1695–1704, 2005.

[33] S. Y. Kung, S.-H. Lin, and M. Fang, "A neural network approach to face/palm recognition," in *Proc. IEEE Workshop Neural Netw. Signal Process.*, Cambridge, MA, USA, 1995, pp. 323–332.

[34] V. Laxmi, "Palmprint matching using LBP," in *Proc. IEEE Int. Conf. Comput. Sci. (ICCS)*, 2012, pp. 110–115.

[35] M. A. L. Vijilious and V. S. Bharathi, "Texture feature extraction approach to palmprint using nonsubsampled contourlet transform and orthogonal moments," *Int. J. Future Comput. Commun.*, vol. 1, no. 3, pp. 298–301, 2012.

[36] M. K. H. Leung, A. C. M. Fong, and S. C. Hui, "Palmprint verification for controlling access to shared computing resources," *IEEE Pervasive Comput.*, vol. 6, no. 4, pp. 40–47, Oct./Dec. 2007.

[37] F. Li, M. K. H. Leung, and X. Yu, "Palmprint identification using Hausdorff distance," in *Proc. Int. Workshop Biomed. Circuits Syst.*, Singapore, 2004, pp. 1–6.

[38] G. Lu, D. Zhang, and K. Wang, "Palmprint recognition using eigenpalms features," *Pattern Recognit. Lett.*, vol. 24, nos. 9–10, pp. 1463–1467, 2003.

[39] Y.-T. Luo *et al.*, "Local line directional pattern for palmprint recognition," *Pattern Recognit.*, vol. 50, pp. 26–44, Feb. 2016.

[40] A. Nigam and P. Gupta, "Palmprint recognition using geometrical and statistical constraints," in *Proc. 2nd Int. Conf. Soft Comput. Problem Solving (SocProS)*, Jaipur, India, 2014, pp. 1303–1315.

[41] J. S. Noh and K. H. Rhee, "Palmprint identification algorithm using Hu invariant moments and Otsu binarization," in *Proc. 4th Annu. ACIS Int. Conf. Comput. Inf. Sci.*, 2005, pp. 94–99.

[42] G. K. O. Michael, T. Connie, and A. B. J. Teoh, "Touch-less palm print biometrics: Novel design and implementation," *Image Vis. Comput.*, vol. 26, no. 12, pp. 1551–1560, 2008.

[43] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[44] D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "A dynamic algorithm for palmprint recognition," in *Proc. IEEE Conf. Commun. Netw. Security (IEEE CNS) 1st Workshop Security Privacy Cybermatics (SPiCy)*, Florence, Italy, 2015, pp. 623–626.

[45] C. Phromsuthirak, S. Suwan, A. Sanpanich, and C. Pintavirooj, "Hand shape identification using palmprint alignment based on intrinsic local affine-invariant fiducial points," in *Proc. Biomed. Eng. Int. Conf. (BMEiCON)*, Fukuoka, Japan, 2014, pp. 1–5.

[46] X. Pin and Q.-Q. Ruan, "Palmprint recognition using Gabor feature-based $(2D)^2$PCA," *Neurocomputing*, vol. 71, nos. 13–15, pp. 3032–3036, 2008.

[47] R. Raghavendra, B. Dorizzi, A. Rao, and G. H. Kumar, "PSO versus adaboost for feature selection in multimodal biometrics," in *Proc. 3rd IEEE Int. Conf. Biometrics Theory Appl. Syst.*, Washington, DC, USA, 2009, pp. 238–244.

[48] S. Ribaric, I. Fratric, and K. Kis, "A biometric verification system based on the fusion of palmprint and face features," in *Proc. 4th Int. Symp. Image Signal Process. Anal.*, Zagreb, Croatia, 2005, pp. 12–17.

[49] R. Sanchez-Reillo, C. Sanchez-Avilla, and A. Gonzalez-Marcos, "Biometric identification through hand geometry measurements," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1168–1171, Oct. 2000.

[50] L. Shengcai, A. K. Jain, and S. Z. Li, "Partial face recognition: Alignment-free approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 5, pp. 1193–1205, May 2013.

[51] Z. Sun, Y. Wang, T. Tan, and J. Cui, "Robust direction estimation of gradient vector field for iris recognition," in *Proc. 17th Int. Conf. Pattern Recognit.*, Cambridge, U.K., 2004, pp. 783–786.

[52] Z. Sun, T. Tan, Y. Wang, and S.Z. Li, "Ordinal palmprint represention for personal identification [represention read representation]," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1. San Diego, CA, USA, 2005, pp. 279–284.

[53] J. A. Wincy and J. G. C. Chandran "Palmprint recognition using PCF and SURF," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 3, no. 10, pp. 996–1001, 2013.

[54] X. Wu, K. Wang, and D. Zhang, "Fuzzy directional element energy feature (FDEEF) based palmprint identification," in *Proc. Int. Conf. Pattern Recognit.*, vol. 1. Quebec City, QC, Canada, 2002, pp. 95–98.

[55] X. Wu, K. Wang, and D. Zhang, "Line feature extraction and matching in palmprint," in *Proc. 2nd Int. Conf. Image Graph.*, 2002, pp. 583–590.

[56] X. Wu, D. Zhang, K. Wang, and B. Huang, "Palmprint classification using principal lines," *Pattern Recognit.*, vol. 37, no. 10, pp. 1987–1998, 2004.

[57] X. Wu, K. Wang, and D. Zhang, *HMMs Based Palmprint Identification* (Lecture Notes in Computer Science), vol. 3072. Berlin, Germany: Springer, 2004, pp. 775–781.

[58] X. Wu, D. Zhang, and K. Wang, "Palm line extraction and matching for personal authentication," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 5, pp. 978–987, Sep. 2006.

[59] J. You, W. Li, and D. Zhang, "Hierarchical palmprint identification via multiple feature extraction," *Pattern Recognit.*, vol. 35, no. 4, pp. 847–859, 2002.

[60] D. Zhang, A. W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041–1050, Sep. 2003.

[61] B. Zhang, W. Li, P. Qing, and D. Zhang, "Palm-print classification by global features," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 43, no. 2, pp. 370–378, Mar. 2003.

[62] D. Zhang and W. Shu, "Two novel characteristics in palmprint verification: Datum point invariance and line feature matching," *Pattern Recognit.*, vol. 32, no. 4, pp. 691–702, 1999.

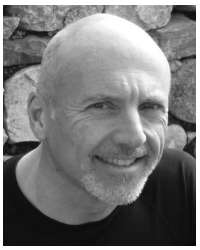
[63] Z.J. Zia-uddin, Z. Jan, J. Ahmad, and A. Abbasi, "A novel technique for principal lines extraction in palmprint using morphological top-hat filtering," *World Appl. Sci. J.*, vol. 31, no. 12, pp. 2010–2014, 2014.

[64] *PolyU Palmprint Database*, Biometric Res. Center Hong Kong Polytech. Univ., Hong Kong, 2003. [Online]. Available: http://www.comp.polyu.edu.hk/∼biometrics/

[65] *CASIA Palmprint Image Database*, Chin. Acad. Sci. Inst. Autom., Beijing, China, 2005. [Online]. Available: http://biometrics.idealtest.org/

David Palma received the B.Sc. degree in information engineering and management and the M.Sc. degree in electronic engineering from the University of Udine, Udine, Italy, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree with the Polytechnic Department of Engineering and Architecture.

From 2014 to 2017, he was with the Department of Electrical, Mechanical and Management Engineering, University of Udine, where he was a Research Assistant in biometric authentication, cybersecurity, and pervasive computing. His current research interests include biometric authentication, pervasive computing, and cybersecurity.



Pier Luca Montessoro was born in Turin, Italy, in 1961. He received the Dr.Eng. degree in electronic engineering from the Polytechnic of Turin, Turin, in 1986.

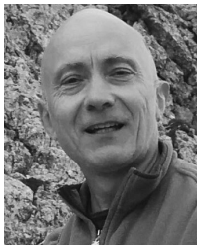
He is currently a Full Professor of computer science with the University of Udine, Udine, Italy. He has been with the Italian National Council for Scientific Research, Rome, Italy, and a Scientific Consultant with Digital Equipment Corporation, Maynard, MA, USA (currently Compaq) in the field of simulation for very large scale integration design. His research interests, after several years spent on CAD systems for digital circuits design and on multimedia systems for e-learning. His current research interests include computer networks, ICT security and pervasive computing, in particular distributed controls and algorithms for agents-based systems.

He has been the Chair and an Organizer of the WCC 2013 workshop "International Workshop on Cloud Convergence: Challenges for Future Infrastructures and Services," hosted in the IEEE ICC conference and chair of the 30th edition of the Didamatica conference, held in Udine.

Giulia Giordano received the B.Sc. and M.Sc. degrees (*summa cum laude*) and the Ph.D. degree from the University of Udine, Udine, Italy, in 2010, 2012, and 2016, respectively.

She was with the Department of Automatic Control, Lund University, Lund, Sweden. She is currently an Assistant Professor with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. She visited the California Institute of Technology, Pasadena, CA, USA, in 2012 and the University of Stuttgart, Stuttgart, Germany, in 2015. Her current research interests include analysis of biological systems and the control of networked systems.

Dr. Giordano was a recipient of the EECI Ph.D. Award 2016 for her doctoral thesis and the NAHS Best Paper Award 2017.



Franco Blanchini was born in Legnano, Italy, in 1959. He received the Laurea degree in electrical engineering from the University of Trieste, Trieste, Italy, in 1984.

He is the Director of the Laboratory of System Dynamics, University of Udine, Udine, Italy. He has co-authored the book entitled *Set Theoretic Methods in Control* (Birkhäuser).

Mr. Blanchini was a recipient of the 2001 ASME Oil and Gas Application Committee Best Paper Award, the 2017 NAHS Best Paper Award, and the 2002 IFAC Prize Survey Paper Award and the High Impact Paper Award 2017 for his article *Set Invariance in Control—A Survey*, *Automatica*, 1999. He was an Associate Editor of *Automatica*, from 1996 to 2006 and the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, from 2012 to 2016. Since 2017, he has been an Associate Editor of *Automatica*. He is a Senior Editor of the IEEE CONTROL SYSTEMS LETTERS. He has been involved in the organization of several international events: he was the Program Vice-Chairman of IEEE CDC-ECC 2005, Seville, Spain, the Program Vice-Chairman of IEEE CDC 2008, Cancun, Mexico, the Program Chairman of IFAC ROCOND 2012, Aalborg, Denmark, and the Program Vice-Chairman of IEEE CDC 2013, Florence, Italy.