Neural Comput & Applic
DOI 10.1007/s00521-014-1570-8

Multispectral palmprint recognition using multiclass projection extreme learning machine and digital shearlet transform

Xuebin Xu · Longbin Lu · Xinman Zhang · Huimin Lu · Wanyu Deng

Received: 12 September 2013 / Accepted: 11 March 2014

Abstract A novel multispectral palmprint recognition method is proposed based on multiclass projection extreme learning machine (MPELM) and digital shearlet transform. Extreme learning machine (ELM) is a novel and efficient learning machine based on the generalized single-hidden-layer feedforward networks, which performs well in classification applications. Many researchers' experimental results have shown the superiority of ELM with classical algorithm: support vector machine (SVM). To further improve the performance of multispectral palmprint recognition method, we propose a novel method based on MPELM in this paper. Firstly, all palmprint images are preprocessed by David Zhang's method. Then, we use image fusion method based on fast digital shearlet transform to fuse the multispectral palmprint images. At last, we use the proposed MPELM classifier to determine the final multispectral palmprint classification. The experimental results demonstrate the superiority of multispectral fusion to each single spectrum, and the proposed MPELM-based method outperforms the SVM-based and ELM-based methods. The proposed method is also suitable for other biometric applications and gets to be work well.

Keywords Extreme learning machine · Multispectral palmprint · Digital shearlet transform · Image fusion

X. Xu · L. Lu · X. Zhang (✉) · W. Deng
School of Electronics and Information Engineering, Xi'an Jiaotong University, 28 Xian'ning West Road, Xi'an 710049, China
e-mail: zhangxinman@mail.xjtu.edu.cn

H. Lu
School of Software Engineering, Changchun University of Technology, Changchun 130012, China

1 Introduction

In recent years, palmprint has become a reliable and unique biometric characteristic with high usability [1]. With the increasing demand of robust and real-time palmprint recognition system, the multispectral palmprint recognition method has been proposed to acquire more discriminative information and increase the palmprint recognition accuracy [2–6].

Some works have been carried on multispectral palmprint recognition. Xu et al. [2] proposed a multispectral palmprint recognition method based on quaternion principal component analysis (QPCA) and quaternion discrete wavelet transform (QDWT). Han et al. [3] presented an online multispectral palmprint system by multispectral image fusion. Rowe et al. [4] designed a multispectral high-resolution whole-hand biometric system, which could collect palmprint images with fingerprint features. Hao et al. [5] developed a contact-less multispectral palmprint device and used image fusion method to integrate multispectral palmprint information. Xu et al. [6] presented a multispectral palmprint recognition method based on principal component analysis (PCA). Zhang et al. [1] designed an online multispectral palmprint system that could meet the requirement of real-time application. A large multispectral palmprint database is established to investigate the recognition performance of each spectral band. The multispectral palmprint images in the database are captured under Blue, Green, Red, and near-infrared (NIR) illuminations. The above works show the superiority of multispectral palmprint systems to the unimodal systems.

There are four levels of fusion in the multispectral palmprint recognition methods: pixel level, feature level, score level, and decision level. To data, most research on

Published online: 30 March 2014

the fusion techniques were studied based on score level and feature level. For example, Zhang et al. [1] proposed a score level fusion scheme to integrate multispectral information. Less work has been done to address pixel level fusion problem.

Recently, a simple and efficient learning algorithm for single-hidden-layer feedforward neural networks (SLFNs) called extreme learning machine (ELM) has been proposed by Huang [8–11]. ELM has been successfully applied to a number of real-world applications [8–17], showing better generalization performance with an extremely fast learning speed. In ELM, input weights and biases can be randomly assigned and the output weights can be analytically determined by the simple generalized inverse operation. Compared with other classical learning machines [8–17], ELM not only learns much faster with higher generalization ability but also avoids many difficulties, such as the stopping criteria, learning rate, learning epochs, and local minima [9]. Due to these advantages, ELM has been successfully applied to many classification and regression problems and has received many researchers' interest [8–17].

Mohammed et al. [12] proposed a new face recognition algorithm based on ELM and bidirectional two dimensional principal component analysis (B2DPCA). The proposed method is based on curvelet image decomposition of human faces and a sub-band that exhibits a maximum standard deviation is dimensionally reduced using an improved dimensionality reduction technique [12]. Other notable contributions of the work include significant improvements in classification rate, up to hundred folds reduction in training time and minimal dependence on the number of prototypes [12]. Extensive experiments are performed using challenging databases and good experimental results are obtained.

ELM classifier overcomes many issues in traditional gradient algorithms such as stopping criterion, learning rate, number of epochs, and local minima. However, ELM has some disadvantages: (1) ELM needs more hidden neurons than BP; (2) The generalization performance of ELM algorithm depends on the proper selection of parameters (number of hidden neuron, input weights) [7–11], especially for fewer number of training samples. It is difficult to find the best parameters such that the training and testing accuracy are to achieve maximum value for a given problem [10–12]. Thus, it is important to address the existing problem of ELM classifier.

In this paper, we propose a novel multispectral palmprint recognition method that operates at the pixel fusion level. Firstly, we propose a novel multispectral palmprint image fusion method based on digital shearlet transform. Compared with classical wavelet transform and curvelet transform, digital shearlet transform has better optimal approximation performance [18–20]. Digital shearlet transform is used to fuse the multispectral palmprint images in this paper. Then, we propose a novel classifier—multiclass projection extreme learning machine (MPELM). MPELM is used to determine the final palmprint image classification.

This paper is organized as follows. Section 1 gives a brief review of the current ELM algorithms. Section 2 describes digital shearlet transform. Section 3 gives out the multispectral palmprint recognition method based on fast digital shearlet transform and MPELM. The experimental results and performance evaluation on Hong Kong Polytechnic University multispectral palmprint database are given in Sect. 4. The last section summarizes the paper and conclusions are drawn finally.

2 Overview of digital shearlet transform

It is widely acknowledged that analyzing the intrinsic geometrical features of the underlying image is essential in many applications including image processing and biometric recognition [18, 20]. In order to achieve this, some transforms have been proposed to capture geometric regularity of a given image. They include curvelet, contourlet and bandelet transforms [21]. In order to achieve better performance than the above transforms, Donoho, Kutyniok, and Wang-Q Lim et al. proposed digital shearlet transform in 2010 [18, 20]. Shearlet transform is a new sparse representation which yields nearly optimal approximation properties. In this section, we briefly review the implementation of fast digital shearlet transform (FDST) which is simpler and faster [18–21]. The main steps are as follows.

2.1 Weighted pseudo-polar transforms

Given an $N \times N$ image I, we need to choose weights $\omega : \Omega_R \rightarrow \mathbb{R}^+$ so that:

$$\sum_{u,v=-N/2}^{N/2-1} |I(u,v)|^2 = \sum_{(\omega_x,\omega_y)\in\Omega_R} \omega(\omega_x,\omega_y).\left|\hat{I}(\omega_x,\omega_y)\right|^2 \quad (1)$$

Here, $\hat{I}(\omega_x,\omega_y)$ is pseudo-polar Fourier transform (PPFT) given by

$$\hat{I}(\omega_x,\omega_y) = \sum_{u,v=-N/2}^{N/2-1} I(u,v)\mathrm{e}^{-\frac{2\pi i}{RN+1}(u\omega_x+v\omega_y)} \quad (2)$$

and $\Omega_R = \Omega_R^1 \cup \Omega_R^2$ is the pseudo-polar grid with

$$\Omega_R^1 = \left\{\left(-\frac{4lk}{RN},\frac{2k}{R}\right) : -\frac{N}{2} \le l \le \frac{N}{2}, -\frac{RN}{2} \le k \le \frac{RN}{2}\right\} \quad (3)$$

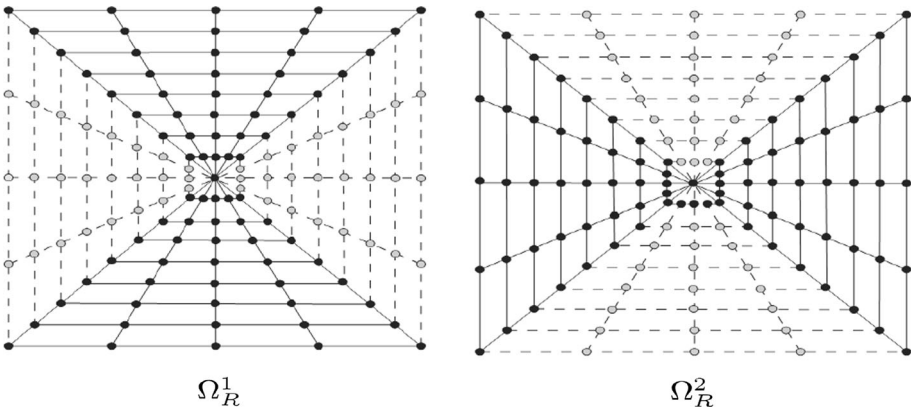

Fig. 1 Pseudo-polar grids ($R = 4$ and $N = 4$)

$$\Omega_R^2 = \left\{ \left(\frac{2k}{R}, -\frac{4lk}{RN} \right) : -\frac{N}{2} \le l \le \frac{N}{2}, -\frac{RN}{2} \le k \le \frac{RN}{2} \right\} \tag{4}$$

and $R \ge 2$ is the oversampling factor. Figure 1 shows these grids. Notice that the center $C = \{(0,0)\}$ appears $N+1$ Times in Ω_R^1 and Ω_R^2, and the points on the seam line $S_R^1 = \left\{ \left(-\frac{2k}{R}, \frac{2k}{R} \right) : -\frac{RN}{2} \le l \le \frac{RN}{2}, k \ne 0 \right\}$ and $S_R^2 = \left\{ \left(\frac{2k}{R}, -\frac{2k}{R} \right) : -\frac{RN}{2} \le l \le \frac{RN}{2}, k \ne 0 \right\}$ appear in both Ω_R^1 and Ω_R^2. The following Plancherel theorem similar to the one for discrete Fourier transform can be proved for pseudo-polar grid $\Omega_R = \Omega_R^1 \cup \Omega_R^2$ by choosing weights ω carefully [17, 19].

Plancherel theorem: Let N be even, $\omega : \Omega_R \to \mathbb{R}^+$ be a weight function. Then, Eq. (1) holds if and only if, for all $-N+1 \le u, v \le N-1$. Weights ω satisfy

$$\begin{aligned} \delta(u,v) = \omega(0,0) &+ 4 \sum_{l=0,N/2} \sum_{k=1}^{RN/2} \omega\left(\frac{2k}{R}, -\frac{4lk}{RN} \right) \\ &\times \cos\left(\frac{2ku}{R(RN+1)} \right) \cos\left(\frac{4lkv}{RN(RN+1)} \right) \\ &+ 8 \sum_{l=1}^{N/2-1} \sum_{k=1}^{RN/2} \omega\left(\frac{2k}{R}, -\frac{4lk}{RN} \right) \\ &\times \cos\left(\frac{2ku}{R(RN+1)} \right) \cos\left(\frac{4lkv}{RN(RN+1)} \right) \end{aligned}$$

and, for all $(\omega_x, \omega_y) \in \Omega_R$, weights ω satisfy the symmetry conditions: $\omega(\omega_x, \omega_y) = \omega(\omega_y, \omega_x), \omega(\omega_x, \omega_y) = \omega(-\omega_x, \omega_y)$ and $\omega(\omega_x, \omega_y) = \omega(\omega_x, -\omega_y)$.

2.2 Recommended choice of weights

To reduce computational cost of weights ω satisfying Plancherel theorem [18, 20, 22], we relax the requirement for exact isometric weighting, however, instead represent the weights not as the solution of a large system of equations, but in terms of an undercomplete basis for functions on the pseudo-polar grid. We compute the coefficients in this expansion once for a given problem size; then hardwire them in the algorithm [19, 21]. For the algorithm, we expand using seven weight functions $(\omega_1, \omega_2, \ldots \omega_7)$ on pseudo-polar grid such that $\sum_{j=1}^{7} \omega_j(\omega_x, \omega_y) \ne 0$ for each $(\omega_x, \omega_y) \in \Omega_R$:

Center: $\omega_1 = 1_{(0,0)}$ and $\omega_2 = 1_{\{(\omega_x,\omega_y):|k|=1\}}$,
Seam lines: $\omega_3(\omega_x, \omega_y) = |k| \cdot 1_{\{(\omega_x,\omega_y):1<|k|=NR/2,\omega_x=\omega_y\}}$ and $\omega_4 = 1_{\{(\omega_x,\omega_y):|k|=NR/2-3,\omega_x=\omega_y\}}$,
Boundary: $\omega_5 = 1_{\{(\omega_x,\omega_y):|k|=NR/2,\omega_x=\omega_y\}}$, and $\omega_6 = 1_{\{(\omega_x,\omega_y):|k|=NR/2,\omega_x\ne\omega_y\}}$,
Interior: $\omega_7(\omega_x, \omega_y) = |k| \cdot 1_{\{(\omega_x,\omega_y):1<|k|=NR/2,\omega_x\ne\omega_y\}}$

It is noticed that here we use (ω_x, ω_y) and (k, l) interchangeably.

2.3 Digital shearlets on the pseudo-polar grid

We can define scaling function ϕ for digital shearlet system to be

$$\hat{\phi}(\xi_1, \xi_2) = W_0(\xi_1) V_0(\xi_2), (\xi_1, \xi_2) \in \mathbb{R}^2. \tag{5}$$

where W_0 is Fourier transform of Meyer scaling function satisfying supp $W_0 \subseteq [-4^{j_R}, 4^{j_R}]$ with $j_R := -\lceil \log_4(R/2) \rceil$, and V_0 is a bump function satisfying supp $V_0 \subseteq [-4^{j_R} - 1/2, 4^{j_R} + 1/2]$ with $V_0(x) \equiv 1$ for $x \le 4^{j_R}$, for which numerous choices exist.

We further choose W to be Fourier transform of Meyer wavelet function satisfying supp $W \subseteq [-4^{j_R+1}, -4^{j_R+1}] \cup [4^{j_R-1}, 4^{j_R+1}]$ and V to be a bump function satisfying supp $V \subseteq [-1, 1]$ and $|V(\xi-1)|^2 + |V(\xi)|^2 + |V(\xi+1)|^2 = 1$ for all $|\xi|^2 \le 1$ and $\xi \in \mathbb{R}$. Then, generating shearlet φ for digital shearlet system on Ω_R^2 is defined as

$$\hat{\varphi}(\xi_1, \xi_2) = W_0(\xi_1) V\left(\frac{\xi_2}{\xi_1} \right), (\xi_1, \xi_2) \in \mathbb{R}^2. \tag{6}$$

Before defining digital shearlets, we can partition the set Ω_R beyond the already defined partitioning into Ω_R^1 and Ω_R^2 by setting $\Omega_R^1 = \Omega_R^{11} \cup C \cup \Omega_R^{12}$ and $\Omega_R^2 = \Omega_R^{21} \cup C \cup \Omega_R^{22}$, where

$$\Omega_R^{11} = \left\{ \left(-\frac{4lk}{RN}, \frac{2k}{R} \right) : -\frac{N}{2} \le l \le \frac{N}{2}, 1 \le k \le \frac{RN}{2} \right\},$$

$$\Omega_R^{12} = \left\{ \left(-\frac{4lk}{RN}, \frac{2k}{R} \right) : -\frac{N}{2} \le l \le \frac{N}{2}, -\frac{RN}{2} \le k \le -1 \right\},$$

$$\Omega_R^{21} = \left\{ \left(\frac{2k}{R}, -\frac{4lk}{RN} \right) : -\frac{N}{2} \le l \le \frac{N}{2}, 1 \le k \le \frac{RN}{2} \right\},$$

$$\Omega_R^{22} = \left\{ \left(\frac{2k}{R}, -\frac{4lk}{RN} \right) : -\frac{N}{2} \le l \le \frac{N}{2}, -\frac{RN}{2} \le k \le -1 \right\},$$

$$\vartheta_j^1 = \begin{cases} 4^{j+jR-1\frac{R}{2}}15 + 1 : 0 \le j < \lceil \log_4 N \rceil - j_R, \\ \left\lfloor \frac{R}{2}(N - 4^{j+jR-1}) \right\rfloor + 1 : j = \lceil \log_4 N \rceil - j_R \end{cases},$$

$$\vartheta_{j,s}^2 = \begin{cases} 2^{-j}N + 1 : -2^j < s < 2^j, \\ 2^{-j}\frac{N}{2} + 1 : s \in \{-2^j, 2^j\} \end{cases},$$

$\Re_{j,s} = \{((\vartheta_j^1)^{-1}4^j(R/2)r_1, -(\vartheta_{j,s}^2)^{-1}$ $(N/2^{j+1})r_2 : \ r_1 = 0, \ldots, \vartheta_j^1 - 1, r_2 = 0, \ldots \vartheta_{j,s}^2 - 1\}$, and $\Re = \left\{(r_1, r_2) : r_1 = -1, \ldots, 1, r_2 = -\frac{N}{2}, \ldots, \frac{N}{2}\right\}$. So digital shearlets can be defined [18, 20]:

Definition of digital shearlets: At scale $j \in \{0, \ldots, \lceil \log_4 N \rceil - j_R\}$, shear $s = \{-2^j, \ldots, 2^j\}$ and spatial position $m \in \Re_{j,s}$, digital shearlets on the cone Ω_R^2 are defined by

$$\sigma_{j,s,m}^{21}(\omega_x, \omega_y) = \frac{C(\omega_x, \omega_y)}{\sqrt{|\Re_{j,s}|}} W(4^{-j}\omega_x) V\left(s + 2^j \frac{\omega_y}{\omega_x}\right) X\Omega_R^{21}(\omega_x, \omega_y) e^{-2\pi i m' \left(4^{-j}\frac{2k}{R}, -2^{j+1}\frac{l}{N}\right)}$$

where

$$C(\omega_x, \omega_y) = \begin{cases} 1 : (\omega_x, \omega_y) \notin S_R^1 \cup S_R^2, \\ \frac{1}{\sqrt{2}} : (\omega_x, \omega_y) \in \left(S_R^1 \cup S_R^2\right) \backslash C \\ \frac{1}{\sqrt{2(N+1)}} : (\omega_x, \omega_y) \in C. \end{cases},$$

The shearlets $\sigma_{j,s,m}^{11}$, $\sigma_{j,s,m}^{12}$, and $\sigma_{j,s,m}^{22}$ on the remaining cones are defined accordingly by symmetry with equal indexing sets. For $n \in \Re$, we further define the function

$$\varphi_n^l(\omega_x, \omega_y) = \frac{C(\omega_x, \omega_y)}{\sqrt{|\Re|}} W_0(\omega_x) V_0(\omega_y) X\Omega_R^{21}(\omega_x, \omega_y) \mathrm{e}^{-in'\left(\frac{k}{3 \cdot N+1}, \frac{l}{N}\right)}, \ l = 1, 2.$$

At last, we call the system

$$\left\{\varphi_n^l : l = 1, 2, n \in \Re\right\} \cup \{\sigma_{j,s,m}^{11}, \sigma_{j,s,m}^{12}, \sigma_{j,s,m}^{21}, \sigma_{j,s,m}^{22} : j \in \{0, \ldots, \lceil \log_4 N \rceil - j_R\}, s = \left\{-2^j, \ldots, 2^j\right\}, m \in \Re_{j,s}\}$$

the digital shearlet transform.

2.4 Implementation of the FDST and its inverse

Figure 2 provides an overview of the main steps of FDST and its inverse. The implementation of FDST consists of three parts: fast PPFT, weighting on the pseudo-polar grid, and digital shearlet, windowing on the pseudo-polar grid followed by 2D-iFFT.

Readers may wish to refer to [19, 20, 22] for more details.

3 Multispectral palmprint recognition algorithm using FDST and MPELM

Figure 3 shows the procedure of multispectral palmprint recognition algorithm using FDST and MPELM.

Firstly, all images should be preprocessed and ROI should be extracted. Then, we use image fusion method based on FDST to fuse the multispectral palmprint images. At last, MPELM is used to determine the final recognition results.

The main procedure of the proposed algorithm is as follows.

3.1 Preprocessing and image normalization

Firstly, we extract region of interesting (ROI) of the palm images. Please refer to Ref. [23–25] for details. Let A and B represent near-infrared (NIR) palmprint and Red/Green/Blue palmprint image samples separately. At last, the gray level of all testing and training images should be scaled to [0, 1], and the image size should be normalized to 64 × 64 pixels.

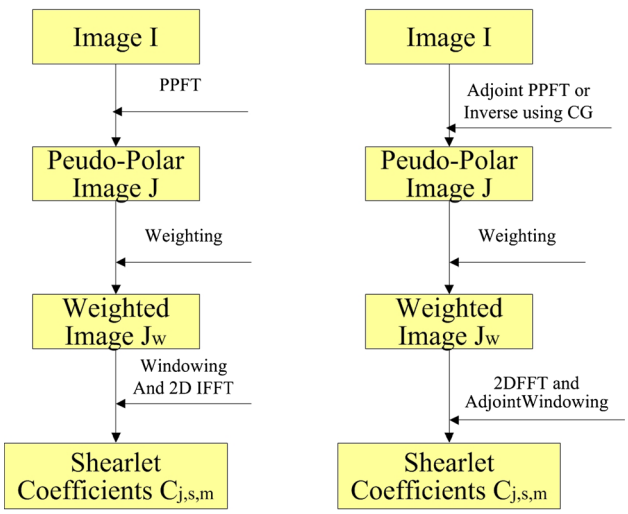

Fig. 2 The main procedure of FDST and its inverse

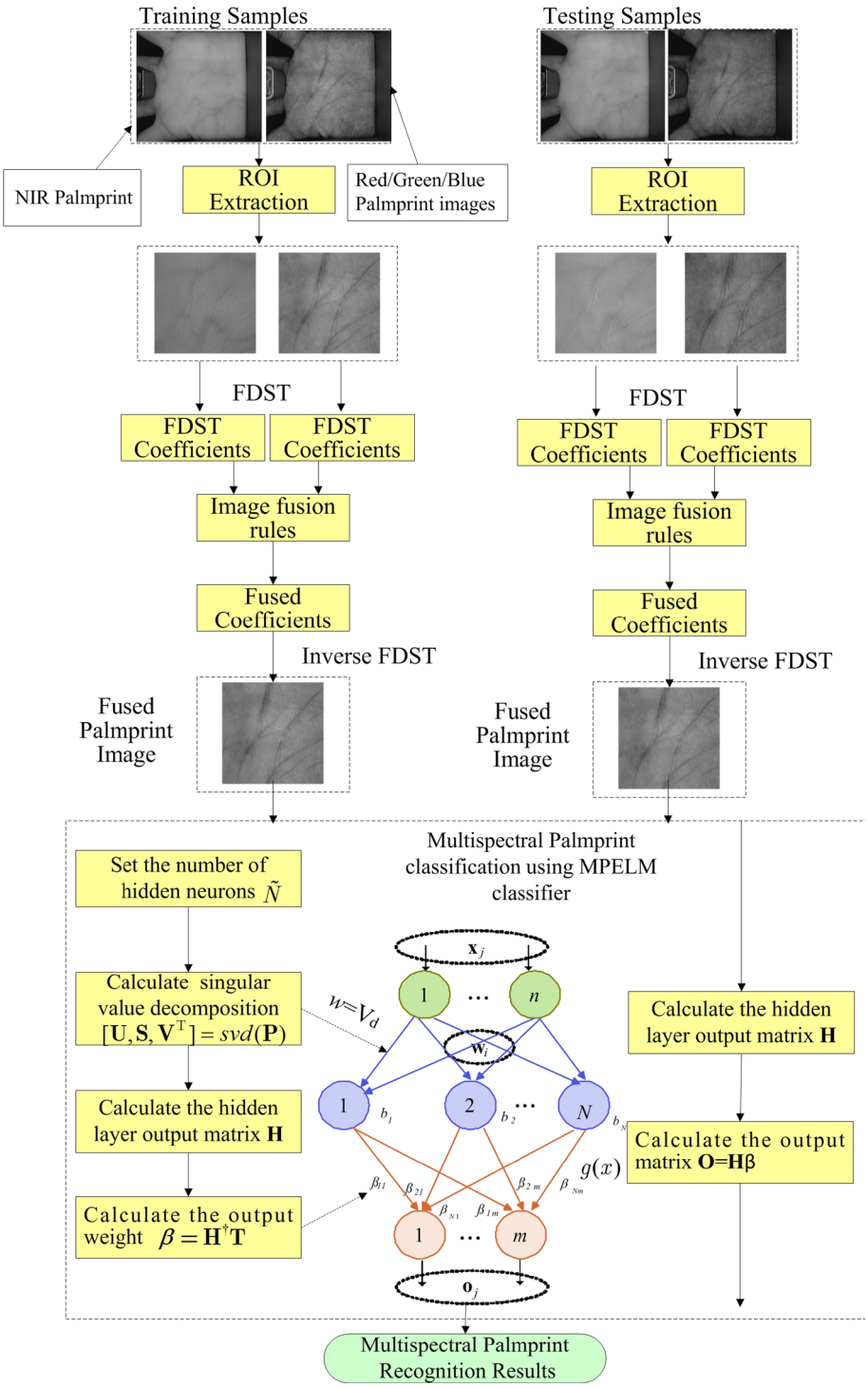

Fig. 3 Schematic of MPELM-based multispectral palmprint recognition method

3.2 Multispectral palmprint image fusion based on fast digital shearlet transform

In this stage, all images are decomposed by using fast digital shearlet transform. In this paper, we use three scales of decomposition for FDST.

Assume that A and B represent NIR palmprint and R/G/B palmprint image sample, respectively.

A and B are decomposed by fast digital shearlet transform into shearlet coefficients, i.e., $\{C_A^{j,s}, C_B^{j,s}\}$, where $C_A^{j,s}, C_B^{j,s}$ represent the coefficients at scale j and shear s.

For pixel level image fusion algorithms, fusion rule design is important. We design the fusion rule for multispectral palmprint image fusion as follows.

$$C_F^{j,s} = \begin{cases} C_A^{j,s} & j = 1 \\ C_B^{j,s} & j > 1 \quad \text{and} \quad abs(C_A^{j,s}) \le abs(C_B^{j,s}) \\ C_A^{j,s} & j > 1 \quad \text{and} \quad abs(C_A^{j,s}) > abs(C_B^{j,s}) \end{cases} \tag{7}$$

where C_F are the fused shearlet coefficients.

Then, we reconstruct the fused image by using inverse fast digital shearlet transform (IFDST).

At last, to reduce computational cost of the classifiers, we resize the size of each fused image to 40 × 40 pixels by bilinear interpolation.

3.3 Multiclass projection extreme learning machine

We propose a novel classifier called MPELM based on the SLFNs in this section [8–12, 14, 15].

Firstly, we define some symbols as follows:

n: Number of samples
m: Dimensions of dataset
k: Label of samples
l: Number of classes
$\tilde{N}$: Number of hidden neurons
$\mathbf{P}_{n\times(m+1)} = \{[\mathbf{x}_k, 1]|\mathbf{x}_k \in \mathbf{R}^m\}_{k=1}^n$: Input in matrix format
$\mathbf{w}_{m\times\tilde{N}}$: Input weights matrix
$\mathbf{Z}_{n\times\tilde{N}}$: Intermediate matrix into hidden layer

$$\mathbf{Z} = \begin{bmatrix} z_{11} & \cdots & z_{1\tilde{N}} \\ \vdots & \vdots & \vdots \\ z_{n1} & \cdots & z_{n\tilde{N}} \end{bmatrix}_{n\times\tilde{N}} = \begin{bmatrix} \mathbf{w}_1 \cdot [\mathbf{x}_1, 1] & \cdots & \mathbf{w}_{\tilde{N}} \cdot [\mathbf{x}_1, 1] \\ \vdots & \vdots & \vdots \\ \mathbf{w}_1 \cdot [\mathbf{x}_n, 1] & \cdots & \mathbf{w}_{\tilde{N}} \cdot [\mathbf{x}_n, 1] \end{bmatrix}_{n\times\tilde{N}}$$

$\mathbf{H}_{n\times\tilde{N}} = \begin{bmatrix} g(z_{11}) & \cdots & g(z_{1\tilde{N}}) \\ \vdots & \vdots & \vdots \\ g(z_{n1}) & \cdots & g(z_{n\tilde{N}}) \end{bmatrix}_{n\times\tilde{N}}$: Output matrix of hidden layer

$\mathbf{T}_{n\times1}$: Target in matrix format $\mathbf{T} = \{\mathbf{t}_k|\mathbf{t}_k \in \mathbf{R}^l\}_{k=1}^n$

For n samples $\{(\mathbf{x}_k, \mathbf{t}_k)\}_{k=1}^n$ where $\mathbf{x}_k = [x_{k1}, x_{k2}, \ldots, x_{km}]$ and $\mathbf{t}_k = [t_{k1}, t_{k2}, \ldots, t_{kl}]$, a standard SLFN with $\tilde{N}$ hidden neurons and activation function $g(\cdot)$ is mathematically modeled as

$$\mathbf{o}_k = \sum_{i=1}^{\tilde{N}} \beta_i g(w_i, \mathbf{x}_k, b_i]), \quad k = 1, \ldots, n \tag{8}$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \ldots, w_{im}]$ is weight vector connecting the i-th hidden neuron and the input neurons, $\boldsymbol{\beta}_i = [\beta_{i1}, \beta_{i2}, \ldots, \beta_{il}]$ is weight vector connecting the ith hidden neuron and the output neurons, $\mathbf{o}_k = [o_{k1}, o_{k2}, \ldots, o_{kl}]$ is the k-th output vector of SLFN, and b_i is the bias of the i-th hidden neuron. Set $\mathbf{w}_i \cdot \mathbf{x}_k$ be the inner product of $\mathbf{w}_i$ and $\mathbf{x}_k$. These n equations can be written compactly as

$$\mathbf{O} = \mathbf{H}\boldsymbol{\beta} \tag{9}$$

where $\mathbf{O} = \begin{bmatrix} \mathbf{o}_1 \\ \vdots \\ \mathbf{o}_n \end{bmatrix}_{n\times m}$, $\mathbf{H} = \begin{bmatrix} g(z_{11}) & \cdots & g(z_{1\tilde{N}}) \\ \vdots & \vdots & \vdots \\ g(z_{n1}) & \cdots & g(z_{n\tilde{N}}) \end{bmatrix}_{n\times\tilde{N}}$,

$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1 \\ \vdots \\ \boldsymbol{\beta}_{\tilde{N}} \end{bmatrix}_{\tilde{N}\times l}$ and $\mathbf{Z} = \begin{bmatrix} z_{11} & \cdots & z_{1\tilde{N}} \\ \vdots & \vdots & \vdots \\ z_{n1} & \cdots & z_{n\tilde{N}} \end{bmatrix}_{n\times\tilde{N}} =$

$$\begin{bmatrix} \mathbf{w}_1 \cdot [\mathbf{x}_1, 1] & \cdots & \mathbf{w}_{\tilde{N}} \cdot [\mathbf{x}_1, 1] \\ \vdots & \vdots & \vdots \\ \mathbf{w}_1 \cdot [\mathbf{x}_n, 1] & \cdots & \mathbf{w}_{\tilde{N}} \cdot [\mathbf{x}_n, 1] \end{bmatrix}_{n\times\tilde{N}}.$$

In order to train an SLFN, one may wish to find specific $\hat{\mathbf{w}}_i$, $\hat{\boldsymbol{\beta}}_i$ $(i = 1, \ldots, \tilde{N})$ such that $||\mathbf{H}(\hat{\mathbf{w}}_1, \ldots, \hat{\mathbf{w}}_{\tilde{N}})\hat{\boldsymbol{\beta}} - \mathbf{T}|| = \min_{\mathbf{w}_i, \boldsymbol{\beta}} ||\mathbf{H}(\mathbf{w}_1, \ldots, \mathbf{w}_{\tilde{N}})\boldsymbol{\beta} - \mathbf{T}||$ which is equivalent to minimizing the cost function

$$E = \sum_{j=1}^{n} \left(\sum_{i=1}^{\tilde{N}} \beta_i g(w_i, \mathbf{x}_k, b_i) - \mathbf{t}_j \right)^2$$

When $\mathbf{H}$ is unknown gradient-based learning algorithms which are generally used to search the minimum of $||\mathbf{H}\boldsymbol{\beta} - \mathbf{T}||$.

Huang et al. proposed a SLFN learning algorithm called the ELM [8, 9, 11]. As rigorously proved by Huang [10, 11], ELM can work as a universal approximator: it is not difficult to find that SLFNs with at most n hidden neurons and with almost any nonlinear activation function which can exactly learn n distinct observations. So, unlike traditional gradient-based learning algorithms, input weights of an SLFN can be randomly chosen (according to any continuous sampling distribution), and output weights of an SLFN can be analytically determined by Moore–Penrose generalized pseudo-inverse [9, 14, 15]. ELM algorithm has been summarized in [9].

In fact, ELM randomly assigns input weights $\mathbf{w}_{(m+1)\times\tilde{N}}$ according to some continuous probability density function. It will cause output weights $\boldsymbol{\beta}$ different in every calculating. Thus, the recognition rate in biometric applications will be changed dynamically. Moreover, ELM needs more hidden neurons than BP [11].

For biometric classification, we propose a novel algorithm called MPELM. MPELM needs less hidden neurons and the recognition rate is explicit in biometric applications.

Firstly, a projection matrix $\mathbf{V}_k \in \mathbf{R}^{m\times k}$ is obtained through singular value decomposition on training dataset

$\mathbf{P}_{n\times m}$, and then we select the k singular vector elements in $\mathbf{V}$ corresponding to largest singular values.

$$[\mathbf{U}, \mathbf{S}, \mathbf{V}^{\mathrm{T}}] = svd(\mathbf{P}) \tag{10}$$

With matrix $\mathbf{V}_k$, we can transform high-dimensional data $\mathbf{P}_{n\times m}$ into low-dimensional data $\mathbf{Z}_{n\times k}$ by

$$\mathbf{Z}_{n\times k} = \mathbf{P}_{n\times m}\mathbf{V}_k \tag{11}$$

Then, $\mathbf{Z}_{n\times k}$ is applied to SLFN to train network. Input weights $\mathbf{w}_{m\times\tilde{N}}$ and output weights $\boldsymbol{\beta}$ of SLFN are determined by learning algorithms such as ELM.

In fact, we need not learn input weights once again. In SLFN, the projection from input layer to hidden layer by input weights $\mathbf{w}_{m\times\tilde{N}}$ can be regarded as dimension reduction,

$$\mathbf{Z}_{n\times\tilde{N}} = \mathbf{P}_{n\times m}\mathbf{w}_{m\times\tilde{N}} \tag{12}$$

$\mathbf{w}_{m\times\tilde{N}}$ can be seen as projection vectors mapping the data into low dimension. So, we use $\mathbf{V}_k \in \mathbf{R}^{m\times k}$ as input weights $\mathbf{w}_{m\times\tilde{N}}$ directly. Obviously, number of hidden neurons $\tilde{N} = k$, the column number of $\mathbf{V}_k$.

MPELM Algorithm: Given a training set $\aleph = \{(\mathbf{x}_k, \mathbf{t}_k)|\mathbf{x}_k \in \mathbf{R}^n, \mathbf{t}_k \in \mathbf{R}^m, k = 1, 2, \ldots, n\}$, an activation function $g(x) = \sin(x)$ or $g(x) = \frac{1}{1+e^x}$.

1. Let $\mathbf{P} = \{[\mathbf{x}_k, 1]|\mathbf{x}_k \in \mathbf{R}^m\}_{k=1}^n$, $\mathbf{T} = \{\mathbf{t}_k|\mathbf{t}_k \in \mathbf{R}^l\}_{k=1}^n$
2. Calculate singular value decomposition of $\mathbf{P}$: $[\mathbf{U}, \mathbf{S}, \mathbf{V}^{\mathrm{T}}] = svd(\mathbf{P})$.
3. Set number of hidden neurons $\tilde{N}$.
4. Set input weights. Let $\mathbf{w} = \mathbf{V}(:, 1:\tilde{N})$.
5. Calculate hidden layer output matrix $\mathbf{H}$.
6. Calculate output weights $\boldsymbol{\beta} = \mathbf{H}^{\dagger}\mathbf{T}$.
7. Input the testing set. Then, we can calculate hidden layer output matrix $\mathbf{H}$ directly.
8. Calculate output matrix $\mathbf{O} = \mathbf{H}\boldsymbol{\beta}$ directly. The classification results will be obtained.

In MPELM algorithm, only output weights need be calculated by least square, input weights have been known after singular value decomposition (SVD). Additionally, there are not iterative learning steps in MPELM algorithm, and thus, its learning speed and recognition speed are very fast. It is also easy to build real-time palmprint recognition system since there are not intermediate data and steps in MPELM algorithm.

4 Experimental results

In our experiments, we use PolyU multispectral palmprint database provided by Hong Kong Polytechnic University [1]. The multispectral palmprint database is a large database which contains 24,000 palmprint images captured with near-infrared (NIR) and visible light (Red, Green, and Blue color). Multispectral palmprint images were collected from 250 volunteers, including 195 males and 55 females [1]. The age distribution is from 20 to 60 years old. These samples were collected in two separate stages. In each stage, the subject was asked to provide 6 images for each palm. Therefore, 12 images of each illumination were collected from each palm. The average time interval between the first and the second stages was about 9 days. For each band (NIR, Red, Green, and Blue), It has a total of $500 \times 12 = 6{,}000$ images obtained from 500 different palms (500 class). Thus, there are $6{,}000 \times 4 = 24{,}000$ palmprint images in the database.

A comparative study among support vector machine (SVM)-based methods, ELM-based methods and MPELM-based methods will be presented in this section. SVM was first developed by Vapnik for pattern recognition [26]. It has also been proved to be very successful in many applications such as iris recognition and face recognition. In our experiments, we use sequential minimal optimization (SMO) SVM for its high performance and least squares (LS) SVM [26–29]. For solving multiclass problem, there are two most popular approaches: One-Against-All (OAA) method and One-Against-One (OAO) method. To reduce the training time and classification time, we use One-Against-All method [30] so 500 binary classifiers need to be constructed. To reduce the computational cost of SVM-based methods, we use locality preserving projection (LPP) method to reduce dimensionality [31]. For MPELM or ELM method, only one multiclass classifier needs to be constructed.

The activation function $g(x) = \sin(x)$ is used in ELM and MPELM classifiers. Because the activation function $g(x) = \sin(x)$ has better performance than other functions in the experiments. The reason why $\sin(x)$ is better than others is that we use singular value decomposition (SVD) in the proposed MPELM method. SVD can reduce the data dimension. The performance of the combination of SVD and the activation function $\sin(x)$ is better than others.

For MPELM and ELM, the nearly optimal number of neurons is selected based on cross-validation method.

4.1 Palmprint recognition on each band

Different bands (NIR, Red/Green/Blue images) can have different features of palm, providing different discriminate information for palmprint recognition [1]. In this section, the recognition results provided by different bands will be reported. We perform the palmprint recognition experiments on the whole database. The image samples are shown in Fig. 4. In the experiments, we randomly choose λ(=1, 2, 3) images of each palm for training. The rest

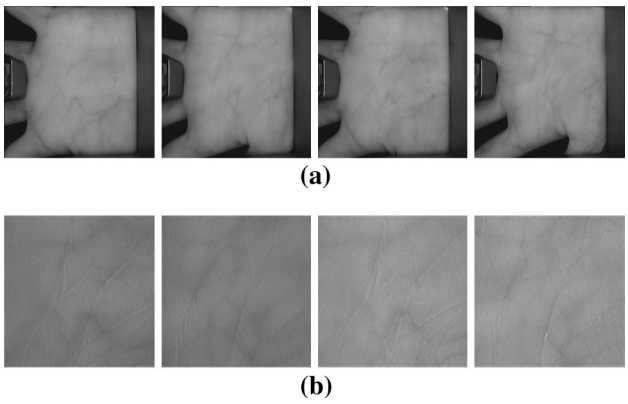

Fig. 4 The image samples from multispectral palmprint database. **a** Original images; **b** ROI region

Table 1 Palmprint recognition results on each band

Methods (Dim)	Average recognition accuracy (%)		
	1 Train	2 Train	3 Train
NIR + MPELM	96.91	98.91	99.21
Red + MPELM	**97.11**	**99.10**	**99.45**
Green + MPELM	96.74	98.53	99.05
Blue + MPELM	95.77	97.96	98.58
NIR + ELM	91.19	96.12	96.87
Red + ELM	91.23	96.38	98.08
Green + ELM	89.49	94.43	95.93
Blue + ELM	87.10	93.28	95.02
NIR + LPP + SMOSVM(100)	87.03	96.89	98.78
Red + LPP + SMOSVM(100)	90.80	97.71	98.21
Green + LPP + SMOSVM(100)	87.01	96.06	97.71
Blue + LPP + SMOSVM(100)	75.93	90.88	96.09
NIR + LPP + LSSVM(100)	86.87	96.57	98.22
Red + LPP + LSSVM(100)	90.11	97.31	97.96
Green + LPP + LSSVM(100)	86.77	95.98	97.45
Blue + LPP + LSSVM(100)	86.65	90.25	95.75

images are considered as the testing datasets. In order to evaluate the performance of our method, for each λ, we have 30 times random selection of the training dataset and take the average recognition rate as the final results. For MPELM and ELM, the number of hidden neurons is selected based on cross-validation method. For SVM-based method, Gaussian radial basis function is used as kernel function and the setting parameters (C and γ) are selected based on cross-validation method. The experimental results are shown in Table 1.

Some findings could be found in Table 1. Firstly, Red and NIR bands have better recognition accuracy than the Blue and Green bands. This is mainly because Red and NIR bands could capture not only most of the palm line but also some vein structures. The vein information helps classify those palms with similar palm lines [1]. Secondly, we can see that MPELM classifier outperforms ELM classifier. Especially, the performance of Red + MPELM method is more better than other methods. Compared with the Red + ELM method, the proposed method (Red + MPELM) can make the average recognition rates increase by 5.88 % (1 Train), 2.72 % (2 Train), 1.37 % (3 Train), respectively. Thirdly, we can see that MPELM classifier outperforms SVM-based classifier. Compared with the Red + LPP + SMOSVM method, the proposed method (Red + MPELM) can make the average recognition rates increase by 6.31 % (1 Train), 1.39 % (2 Train), 1.24 % (3 Train), respectively. At last, MPELM classifier is suitable for single sample recognition application [24]. Even using 1 training sample, better experimental results have been obtained by MPELM classifier. For example, the proposed method (Red + MPELM) obtained 97.11 % (1 Train) accuracy in our experiments.

4.2 Palmprint recognition by multispectral image fusion

The different bands can be fused to further improve the palmprint recognition accuracy. Firstly, we use FDST-based image fusion method to fuse multispectral palmprint images. In the experiments, we use the fused NIR + Red, NIR + Green, NIR + Blue images. There are $6{,}000 \times 4 = 24{,}000$ palmprint images in the database. Each band has 6,000 images. At last, 18,000 fused images are obtained by using FDST-based image fusion method. The fused-image samples are shown in Fig. 5.

In our experiments, we randomly choose λ(=1, 2, 3) fused images of each palm for training. For MPELM and ELM methods, the neuron numbers are selected based on cross-validation method. Gaussian radial basis function is used as kernel function and the setting parameters (C and γ) are selected based on cross-validation method. The experimental results are shown in Table 2.

In Table 2, NIR + Red + MPELM method achieves the best recognition accuracy of 97.33 % (1 Train), 99.22 % (2 Train), and 99.56 % (3 Train). Compared with

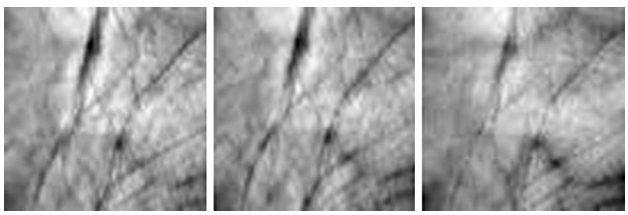

Fig. 5 The fused-image samples (NIR + Red, NIR + Green, NIR + Blue images)

Table 2 Multispectral palmprint recognition results by fusion

Methods (Dim)	Average recognition accuracy (%)		
	1 Train	2 Train	3 Train
NIR + Red + MPELM	97.33	99.22	99.56
NIR + Green + MPELM	97.33	99.01	99.51
NIR + Blue + MPELM	97.05	98.95	99.17
NIR + Red + ELM	91.79	97.34	98.41
NIR + Green + ELM	91.39	96.06	97.98
NIR + Blue + ELM(100)	90.85	95.91	97.46
NIR + Red + LPP + SMOSVM(100)	92.59	98.27	99.21
NIR + Green + LPP + SMOSVM(100)	90.15	97.54	99.05
NIR + Blue + LPP + SMOSVM(100)	88.44	97.30	98.62
NIR + Red + LPP + LSSVM(100)	92.21	98.08	98.93
NIR + Green + LPP + LSSVM(100)	90.01	97.32	98.51
NIR + Blue + LPP + LSSVM(100)	88.17	97.07	98.38

NIR + MPELM method, the proposed fusion-based method (NIR + Red + MPELM) can make the average recognition rates increase by 0.42 % (1 Train), 0.31 % (2 Train), 0.35 % (3 Train), respectively. Compared with Red + MPELM method, the proposed fusion-based method (NIR + Red + MPELM) can make the average recognition rates increase by 0.22 % (1 Train), 0.12 % (2 Train), 0.11 % (3 Train), respectively. For the fused NIR + Green and NIR + Blue images, the proposed MPELM-based method also achieves better results than NIR, Green, and Blue images.

Compared with NIR + LPP + SMOSVM method, the proposed fusion-based method (NIR + Red + MPELM) can make the average recognition rates increase by 10.30 % (1 Train), 2.33 % (2 Train), 0.78 % (3 Train), respectively. Compared with Red + LPP + SMOSVM method, the proposed fusion-based method (NIR + Red + MPELM) can make the average recognition rates increase by 6.53 % (1 Train), 1.49 % (2 Train), 1.31 % (3 Train), respectively.

In addition, we can also find that MPELM classifier outperforms ELM classifier, SMO-SVM classifier and LS-SVM classifier. SMO-SVM classifier also outperforms LS-SVM classifier.

4.3 Speed

Computational speed of the different methods is reported in this section. Table 3 shows total fusion time, training time, total classification time, and per image classification time using ELM-based methods, SVM-based methods and MPELM-based methods. The SVD runtime is included in the proposed MPELM method. All of the algorithms are implemented by MATLAB 7.12(2011a, 64bit) and performed on the same computer (Intel Core i7-2600 3.4 GHz CPU, 4-Core, 12 GB RAM, Windows 7-64bit). All of the experiments are conducted in the same development environment. One training sample of each palm is used in the experiments. Fast training and testing speed are offered by MPELM and ELM classifiers enabled us to repeat the experiments many times [10]. Each experiment is executed 100 times and the average results are reported.

We can see from Table 3 that MPELM-based methods are faster than SVM-based methods. For example, training time of Red + LPP + SMOSVM method is 7.1939s, while that of Red + MPELM is 0.8814 s. Red + MPELM method is **8.16** times faster than Red + LPP + SMOSVM method. The average classification time of Red + LPP + SMOSVM method is 4.48 ms, while that of Red + MPELM is 0.09 ms. Red + MPELM method is **50** times faster than Red + LPP + SMOSVM method.

We can also see from Table 3 that MPELM-based methods are faster than ELM-based methods. For example, training time of Red + ELM method is 2.3664 s, while that of Red + MPELM is 0.8814 s, decreasing by 62.75 %. The average classification time of Red + ELM method is 1.94 ms, while that of Red + MPELM is 0.09 ms, decreasing by 95.36 %. Training time of NIR + Red + ELM method is 2.3341 s, while that of NIR + Red + MPELM is 0.9192 s, decreasing by 60.61 %. It is clear that the MPELM-based methods achieve the best performance.

In both the training speed and the classification speed, MPELM-based methods are faster than SVM-based methods and ELM-based methods. So, the conclusion can be drawn that our method needs less computational time and obtains better accuracy than SVM-based methods and ELM-based methods.

In addition, image fusion time is needed in multispectral palmprint recognition methods. For example, fusion time of a NIR palmprint and Red color palmprint image is about 695.10/6,000 = 0.1158 s.

4.4 Some existing methods

A multispectral palmprint recognition method based principal component analysis (PCA) has been proposed to achieve better performance in [6]. The experiments also used PolyU multispectral palmprint database. The best accuracy 81.9737 % has been obtained in the study.

In 2012, Xu et al. [2] proposed a novel multispectral palmprint recognition method based on QPCA and QDWT. Their experiments have been performed on PolyU multispectral palmprint database. A total of 3,000 samples are used as training samples and 3,000 samples are used as

Table 3 Comparison of execution time

Methods (Dim)	Execution time			
	Total fusion time (s)	Training time (s)	Total classification time (s)	Average classification time (ms)
NIR + MPELM		0.8926	0.5362	0.09
Red + MPELM		0.8814	0.5317	0.09
Green + MPELM		0.9248	0.5430	0.10
Blue + MPELM		0.9135	0.5499	0.10
NIR + ELM		2.3865	10.6402	1.93
Red + ELM		2.3664	10.6865	1.94
Green + ELM		2.2906	10.4801	1.91
Blue + ELM		2.3004	10.3616	1.89
NIR + LPP + SMOSVM(100)		6.6912	23.5832	4.28
Red + LPP + SMOSVM(100)		7.1939	24.6932	4.48
Green + LPP + SMOSVM(100)		7.1755	24.9309	4.53
Blue + LPP + SMOSVM(100)		7.7130	24.8305	4.51
NIR + LPP + LSSVM(100)		10.2311	25.6411	4.66
Red + LPP + LSSVM(100)		10.5112	26.8912	4.89
Green + LPP + LSSVM(100)		10.5521	26.8212	4.87
Blue + LPP + LSSVM(100)		10.8121	26.8298	4.87
NIR + Red + MPELM	695.10	0.9192	0.5463	0.10
NIR + Green + MPELM	671.49	0.9156	0.5524	0.10
NIR + Blue + MPELM	681.22	0.9143	0.5424	0.09
NIR + Red + ELM	695.10	2.3341	10.4535	1.90
NIR + Green + ELM	671.49	2.2806	10.4987	1.91
NIR + Blue + ELM	681.22	2.3091	10.3689	1.89
NIR + Red + LPP + SMOSVM(100)	695.10	6.6830	24.0844	4.37
NIR + Green + LPP + SMOSVM(100)	671.49	7.2272	24.8603	4.52
NIR + Blue + LPP + SMOSVM(100)	681.22	7.1426	24.9431	4.53
NIR + Red + LPP + LSSVM(100)	695.10	10.1127	25.8123	4.69
NIR + Green + LPP + LSSVM(100)	671.49	10.4928	26.5121	4.82
NIR + Blue + LPP + LSSVM(100)	681.22	10.7124	26.6239	4.84

testing samples. The QPCA + QDWT method obtained the best accuracy 98.83 % in their experiments [2]. The average classification time is 0.42 ms. It is clear that our method needs less classification time and obtains better accuracy than QPCA + QDWT method.

5 Conclusions

In this paper, we present a multispectral palmprint recognition method based on fast digital shearlet transform (FDST) and MPELM We can draw a number of conclusions.

1. In this paper, we propose a novel multispectral palmprint image fusion method based on digital shearlet transform. Experimental results show that the performance of multispectral palmprint fusion method is better than the classical unimodal methods.
2. We propose a novel multispectral palmprint recognition method based on the MPELM. MPELM-based method needs less computational time and obtains better accuracy than ELM-based methods and SVM-based methods. Experimental results demonstrate effectiveness and robustness of the proposed recognition method. In conclusion, MPELM is a novel and effective classifier for biometric applications.

Acknowledgments The work is supported by the grants from the National Natural Science Foundation of China (No. 0902025), the special financial grant from China Postdoctoral Science Foundation (2012T50807), the grants from China Postdoctoral Science Foundation (No. 20110491661), the grants from the Natural Science Foundation Funded Project of Jilin Province (No. 20130101055JC), the grants from the Industrial Technology Research and Development Special Project of Jilin Province (No. 2011006-9). the grants of The National Development and Reform Commission Project (No. CNGI-05-17-1T) and Huawei Innovation Research Program. Nanyang Technological University is thanked for providing ELM source code

used in this study. Hong Kong Polytechnic University is thanked for providing multispectral palmprint database used in this study.

References

1. Zhang D, Guo Z, Lu G, Zhang L, Zuo W (2010) An online system of multispectral palmprint verification. IEEE Trans Instrum Measurement 59(2):480–490
2. Xu X, Guo Z, Song C, Li Y (2012) Multispectral palmprint recognition using a quaternion matrix. Sensors 12:4633–4647
3. Rowe RK, Uludag U, Demirkus M, Parthasaradhi S, Jain AK (2007) A multispectral whole-hand biometric authentication system. In: Proceedings of the biometric symposium, biometric consortium conference. Baltimore, MD, pp 1–6
4. Han D, Guo Z, Zhang D (2008) Multispectral palmprint recognition using wavelet-based image fusion. Int Conf Signal Process 2008:2074–2077
5. Hao Y, Sun Z, Tan T, Ren C (2008) Multispectral palm image fusion for accurate contact-free palmprint recognition. ICIP 2008:281–284
6. Xu Y, Zhu Q (2011) PCA-based Multispectral band compression and multispectral palmprint recognition. In: 2011 international conference on hand-based biometrics (ICHB), pp 1–4
7. Meraoumia A, Chitroub S, Bouridane A (2011) Fusion of multispectral palmprint images for automatic person identification. In: Proceedings of the Saudi international electronics, communications and photonics conference (SIECPC), pp 1–6
8. Huang G-B, Wang D (2011) Advances in extreme learning machines (ELM2010). Neurocomputing 74(16):2411–2412
9. Huang G-B, Zhu Qin-Yu, Siew C-K (2006) Extreme learning machine: theory and applications. Neurocomputing 70:489–501
10. Huang G-B, Chen L (2008) Enhanced random search based incremental extreme learning machine. Neurocomputing 71:3460–3468
11. Huang G, Chen L (2007) Convex incremental extreme learning machine. Neurocomputing 70(16–18):3056–3062
12. Mohammed AA, Minhas R, Jonathan Wu QM (2011) Human face recognition based on multidimensional PCA and extreme learning machine. Pattern Recogn 44(10–11):2588–2597
13. Deng W, Zheng Q, Lian S (2010) Ordinal extreme learning machine. Neurocomputing 74(3):447–456
14. Cao J, Lin Z, Huang G-B (2012) Voting based extreme learning machine. Inf Sci 185(1):66–77
15. Deng W, Zheng Q, Lian S, Chen L (2012) Projection vector machine: one-stage learning algorithm from high-dimension small-sample data. In: The 2010 international joint conference on neural networks (IJCNN), pp 1–8
16. Zong W, Huang G-B (2011) Face recognition based on extreme learning machine. Neurocomputing 74(16):2541–2551
17. Wang Y, Cao F, Yuan Y (2011) A study on effectiveness of extreme learning machine. Neurocomputing 74(16):2483–2490
18. Donoho DL, Kutyniok G, Shahram M, Zhuang X. A rational design of digital shearlet transform. http://www.home.uni-osnabrueck.de/kutyniok/papers/ShearLab_SampTA.pdf
19. Easley G, Labate D, Lim WQ (2006) Sparse directional image representations using the discrete shearlet transform. In: Proceedings of conference on signals, systems and computers, ACSSC, pp 974–978
20. Lim W-Q (2010) The discrete shearlet transform: a new directional transform and compactly supported shearlet frames. IEEE Trans Image Process 19(5):1166–1180
21. Pennec E, Mallat S (2005) Sparse geometric image representation with bandelets. IEEE Trans Image Process 14(4):423–438
22. Kutyniok G, Lim W-Q (2011) Compactly supported shearlets are optimally sparse. J Approx Theory 163(11):1564–1589
23. Zhang D, Kong W-K, You J, Wong M (2003) Online palmprint identification. IEEE Trans Pattern Anal Mach Intell 25(9):1041–1049
24. Jia W, Huang DS, Zhang D (2008) Palmprint verification based on robust line orientation code. Pattern Recogn 41(5):1504–1513
25. Yao YF, Jing XY, Wong HS (2007) Face and palmprint feature level fusion for single sample biometrics recognition. Neurocomputing 70(7–9):1582–1588
26. Vapnic V (1998) Statistical learning theory. Wiley, New York
27. Lopez J, Dorronsoro JR (2012) Simple proof of convergence of the SMO algorithm for different SVM variants. IEEE Trans Neural Netw Learn Syst 23(7):1142–1147
28. Kuan T-W, Wang J-F, Wang J-C (2012) VLSI design of an SVM learning core on sequential minimal optimization algorithm. IEEE Trans Very Large Scale Integr Syst 20(4):673–683
29. Cai F, Cherkassky V (2012) Generalized SMO algorithm for SVM-based multitask learning. IEEE Trans Neural Netw Learn Syst 23(6):997–1003
30. Chasanis V, Likas A, Galatsanos N (2009) Simultaneous detection of abrupt cuts and dissolves in videos using support vector machines. Pattern Recogn Lett 30(1):55–65
31. He X, Yan S, Hu Y et al (2005) Face recognition using Laplacian faces. IEEE Trans Pattern Anal Mach Intell 27:328–340