The Multiscale Competitive Code via Sparse Representation for Palmprint Verification

Wangmeng Zuo
Harbin Institute of Technology
Harbin, 150001, China
wmzuo@hit.edu.cn

Zhouchen Lin
Microsoft Research Asia
Beijing, 100190, China
zhoulin@microsoft.com

Zhenhua Guo, David Zhang
Hong Kong Polytechnic University
Kowloon, Hong Kong
csdzhang@comp.polyu.edu.hk

Abstract

Palm lines are the most important features for palmprint recognition. They are best considered as typical multiscale features, where the principal lines can be represented at a larger scale while the wrinkles at a smaller scale. Motivated by the success of coding-based palmprint recognition methods, this paper investigates a compact representation of multiscale palm line orientation features, and proposes a novel method called the sparse multiscale competitive code (SMCC). The SMCC method first defines a filter bank of second derivatives of Gaussians with different orientations and scales, and then uses the l_1-norm sparse coding to obtain a robust estimation of the multiscale orientation field. Finally, a generalized competitive code is used to encode the dominant orientation. Experimental results show that the SMCC achieves higher verification accuracy than state-of-the-art palmprint recognition methods, yet uses a smaller template size than other multiscale methods.

1. Introduction

With the increasing demand for biometric solutions to security systems, recently palmprint recognition, a relatively novel and effective biometric technology, has received considerable interest [29]. Palmprint, the inner surface of the palm, carries a variety of distinctive discriminative features, such as geometry, shape, principal lines, wrinkles, and patterns of ridges, which can be easily captured by appropriate imaging sensors. With the progress in sensor techniques, various palmprint recognition technologies, *e.g*. low resolution [29], latent [13], multispectral [25], and 3D [30], have been investigated.

Palmprint recognition methods can be roughly grouped into three categories: holistic [23], local feature-based [18], and hybrid approaches [20]. Among them, coding-based methods, which encode the response of a bank of filters, have been very successful in palmprint representation and matching. After convolving the palmprint image with 2D Gabor filters, the PalmCode method [18, 29] encodes the phase of the filter responses as binary features. Subsequently, to reduce the spatial correlation of PalmCode, Kong *et al*. [16] further introduce a FusionCode method, which encodes the phase of the filter response whose magnitude is the maximum.

Compared to the phase of the filter response, the orientation of palm lines carries more discriminative information for personal identification. To date, most state-of-the-art palmprint recognition approaches, such as the competitive code [17], the ordinal code [26], and the robust line orientation code (RLOC) [14], are orientation coding methods, which involve three components: the filter bank design for extracting the palmprint orientation information, the coding scheme for compact and efficient representation of orientation information, and the matching approach for fast and accurate palmprint recognition. Rather than directly encoding the dominant orientation, Guo *et al*. [8] suggested a binary orientation co-occurrence vector (BOCV) method to represent multiple orientations for a local region.

Besides phase and orientation features, Chu *et al*. [5] proposed a Gabor magnitude feature representation method where AdaBoosting and LDA are adopted to reduce the feature dimensionality. Phase, orientation, or magnitude features can be regarded as some kind of local texture representation method. Thus effective palmprint representation can be derived by developing appropriate texture descriptors. Most recently, a number of local texture descriptors, *e.g*. local direction histogram [12] and regional appearance correlation (RAC) [11], have been proposed for palmprint recognition with high accuracy.

Palm lines are generally considered as typical multiscale features, where the principal lines can be represented at a larger scale while the wrinkles at a smaller scale. Due to the influence of lighting and aging, some wrinkles in the palmprint image may appear or disappear, while the principal lines are robust. Most coding-based methods, however, neglect the multiscale characteristic of palm lines and en-

code the responses of filters at only one specific scale. At a specific scale, although one can get a compact and effective code, the performance of a coding-based method would deteriorate when applied to matching palmprint images of poor quality.

To address this problem, several multiscale palmprint recognition methods have been proposed. At the beginning, multiscale methods are usually adopted to facilitate fast palmprint identification by using the hierarchical matching scheme [22, 27]. Most recently, several approaches have been suggested to improve the recognition performance by utilizing multiscale palmprint features [10, 31]. Zuo *et al*. [31] proposed a combined angular distance to combine the distances of competitive codes [17] obtained at different scales. Han *et al*. [10] proposed an LDA-based method to combine the distances of appearance statistics at different scales.

In this paper, we propose a novel multiscale palmprint verification method, called the sparse multiscale competitive code (SMCC). The proposed SMCC method first defines a filter bank of second derivatives of Gaussians with different orientations and scales, and then uses the l_1-norm sparse coding to obtain a robust estimation of the multiscale/multiorientation filter responses. Finally, the competitive code [17] is extended to encode the dominant orientation of the filter responses. The performance of SMCC on two popular palmprint databases, PolyU and CASIA, is better than other state-of-the-art mono-scale and multiscale palmprint verification methods. The novelty of SMCC includes:

- It uses the sparse representation to robustly estimate the local orientation of palm lines. We model the palmprint representation problem as a sparse coding solution based on known dictionaries, which can be efficiently solved by well-studied convex optimization methods. In contrast, the traditional palmprint recognition methods obtain the response of filters by convolving the filters with the palmprint image, which is less robust when noise is present.

- It encodes the computed sparse codes into one unified code by extending the competitive rule used in the competitive code [17]. Thus it is a compact representation of multiscale features. In comparison, previous multiscale palmprint recognition methods usually use multiple feature vectors to represent multiscale features. This kind of method improves the representation capability at the expense of increasing the template size. Compared with other multiscale coding-based methods, the SMCC has the smallest template size.

The remainder of this paper is organized as follows: Section 2 introduces the sparse palmprint representation by using l_1-norm sparse coding. Section 3 describes the multiscale competitive code via the sparse representation method, and Section 4 presents the experimental results on the PolyU and the CASIA palmprint databases. Finally, Section 5 concludes our paper.

2. 2. Sparse Palmprint Representation

Palm lines are typical multiscale features, where multiscale / multiorientation filter banks could be adopted for multiscale palmprint representation. Given a filter bank (dictionary) $\boldsymbol{D} = [\boldsymbol{d}_1, ..., \boldsymbol{d}_K] \in \mathcal{R}^{n \times K}$ and an image patch $\boldsymbol{x} \in \mathcal{R}^n$, the convolution method is usually used to calculate the filter coefficients, but it suffers from non-orthogonality and crosstalk among the basis filters.

As a palm line patch generally has a specific orientation and scale, the filter coefficients are thus expected to be sparse, i.e., with only few nonzero values. Inspired by recent progress and success in sparse representation, we model the mutiscale palmprint representation problem as a sparse code learning problem with the l_0 regularization,

$$P0: \ \min \|\boldsymbol{\alpha}\|_0, s.t. \ \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \leq \varepsilon, \boldsymbol{\alpha} \in \mathcal{R}^K, \quad (1)$$

where $\|\cdot\|_0$ denotes the l_0-norm, which is the number of nonzero entries in a vector. Generally speaking, sparse code learning involves two sub-problems: dictionary learning and sparse coding [1, 19, 24]. In dictionary learning, a set of training images is used to learn a dictionary to describe the image characteristics. In sparse coding, the sparse representation of an image patch is obtained by solving a sparsity regularized approximation problem.

In our palmprint verification scheme, we utilize both the sparse representation and the multiscale / multiorientation methods to derive an effective palmprint representation method. We use a set of existing second derivatives of Gaussian (sDoG) filters as the dictionary and use the sparse coding method for sparse palmprint representation. The reason for choosing the existing filters rather than a learnt dictionary is that sDoG filters have an explicit physical meaning: they are directly related to the local orientation and scale of the palm lines. So sDoG filters could be utilized to derive a compact and effective multiscale palmprint representation described in Section 3.

2.1. The Filter Bank

The filter bank consists of 18 sDoG filters at three scales and six orientations. The sDoG filter is defined as

$$F(\delta_x, \delta_y, \theta) = \left((x')^2 - \delta_x^2\right) \frac{A}{\delta_x^4} \exp\left(-\frac{(x')^2}{2\delta_x^2} - \frac{(y')^2}{2\delta_y^2}\right), \quad (2)$$

where A is a constant, $x' = (x - x_0)\cos\theta + (y - y_0)\sin\theta$, $y' = -(x - x_0)\sin\theta + (y - y_0)\cos\theta$, (x_0, y_0) is the center

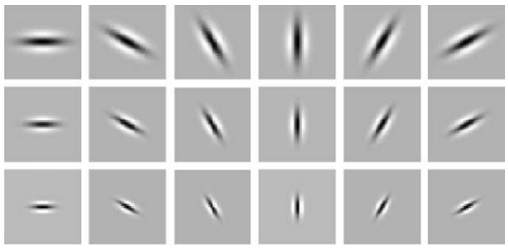

Figure 1. The filter bank used for sparse palmprint representation, which consists of second derivatives of Gaussian filters at three scales and six orientations.

of the filter, θ is the orientation of the filter, and δ_x and δ_y are the horizontal and vertical standard deviations, respectively. Here we choose the number of scales $s = 3$ and the number of orientations $t = 6$. There are only three parameters to determine the filter bank: average $\bar{\delta}_x$ of δ_x, the ratio $\eta = \delta_y/\delta_x$ between δ_y and δ_x, and the scale difference Δ. Then the filter bank parameters are determined by $(\bar{\delta}_x - \Delta, \eta\bar{\delta}_x - \eta\Delta)$, $(\bar{\delta}_x, \eta\bar{\delta}_x)$, and $(\bar{\delta}_x + \Delta, \eta\bar{\delta}_x + \eta\Delta)$, respectively. We first determine the values of $\bar{\delta}_x = 2.7$ and $\eta = 2.5$, and then experimentally determine the value of Δ. The scale parameters for (δ_x, δ_y) are (2.4, 6.0), (2.7, 6.75), and (3.0, 7.5), and the window size of the filter is 33×33. The filter bank used for sparse palmprint representation is shown in Figure 1. Then each filter is preprocessed so that it has a mean of zero and l_1 norm of 1. Finally, all the filters are used to construct the dictionary $\boldsymbol{D}$ for sparse palmprint representation.

Besides sDoG, there are other families of filter bank, such as Gabor [17] and elliptical Gaussian [26], which can also be used for our purpose. Since all these filters are designed to describe line features, it is expected that they will achieve similar performance for orientation coding methods [28]. So in this paper we do not consider the sparse representation by using Gabor or elliptical Gaussian filters.

2.2. Sparse Coding

Given the dictionary $\boldsymbol{D}$ described above, for each patch $\boldsymbol{x}$ of the palmprint image, a sparse solution can be obtained by solving the optimization problem defined in (1). However, this is a nonconvex problem and cannot be efficiently solved. In practice, convex relaxation is usually adopted to change Problem $P0$ into a convex optimization problem by replacing the l_0 regularization with the l_1 regularization,

$$P1: \ \min \|\boldsymbol{\alpha}\|_1, s.t. \ \|\boldsymbol{x} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 \leq \varepsilon, \boldsymbol{\alpha} \in \mathcal{R}^K. \quad (3)$$

Suppose $\boldsymbol{D}$ obeys a uniform uncertainty principle. If the solution to $P0$ is sufficiently sparse, then the solution to $P1$ is guaranteed to stably recover the solution to $P0$ [4]. Problem $P1$ can be formulated into a second-order cone programming problem and be efficiently solved by a number of optimization methods, such as the interior point method [15] and the iteratively reweighted least-squares method [7].

Given a palmprint image $\boldsymbol{I}$, the patch $\boldsymbol{x}_{ij}$ is defined as a subimage with a size of 33×33 and centered at (i, j). We preprocess the patch to make it have a mean of zero and standard deviation of one. Then the sparse code $\boldsymbol{\alpha}_{ij}$ of the patch image is obtained by solving Problem $P1$.

In our method, we first transform Problem $P1$ into an unconstrained optimization problem,

$$P2: \ \min_{\boldsymbol{\alpha}_{ij}} \|\boldsymbol{x}_{ij} - \boldsymbol{D}\boldsymbol{\alpha}_{ij}\|_2^2 + \lambda\|\boldsymbol{\alpha}_{ij}\|_1. \quad (4)$$

By choosing an appropriate λ value, Problem $P2$ has the same solution as Problem $P1$. However, the l_1-regularization is nondifferentiable when $\boldsymbol{\alpha}_{ij}$ contains zero entries. So far, a number of approaches have been developed to solve $P2$ by using sub-gradient [6], unconstrained approximations [21], or constrained optimization strategies [15].

In sparse palmprint representation, we use the recently developed fast iterative shrinkage-thresholding (FISTA) algorithm [2] to solve the unconstrained optimization problem in (4) by introducing a new vector $\boldsymbol{y}_k$. Let $\boldsymbol{a}_k$ be the current estimate of $\boldsymbol{\alpha}_{ij}$ and L denote the Lipschitz constant of the gradient of the function $f(\boldsymbol{\alpha}) = \|\boldsymbol{x}_{ij} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2$. FISTA [2] uses the following quadratic approximation of $\|\boldsymbol{x}_{ij} - \boldsymbol{D}\boldsymbol{\alpha}\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1$ at a given point $\boldsymbol{y}_k$:

$$Q_L(\boldsymbol{\alpha}, \boldsymbol{y}_k) = \begin{array}{l} f(\boldsymbol{y}_k) + \langle \boldsymbol{\alpha} - \boldsymbol{y}_k, \nabla f(\boldsymbol{y}_k)\rangle \\ + \frac{L}{2}\|\boldsymbol{\alpha} - \boldsymbol{y}_k\|_2^2 + \lambda\|\boldsymbol{\alpha}\|_1 \end{array}. \quad (5)$$

By solving the following minimization problem,

$$\boldsymbol{a}_{k+1} = \min_{\boldsymbol{\alpha}} Q_L(\boldsymbol{\alpha}, \boldsymbol{y}_k), \quad (6)$$

we can have an updated solution.

Since the l_1-norm is separable, the computation of $\boldsymbol{a}_{k+1}$ can be efficiently done by solving one 1D minimization problem for each of its components:

$$\boldsymbol{a}_{k+1} = \mathcal{T}_{\lambda/L}\left(\boldsymbol{y}_k - \nabla f(\boldsymbol{y}_k)/L\right), \quad (7)$$

where $\mathcal{T}_a$ is the shrinkage operator defined by

$$\mathcal{T}_a(\boldsymbol{x})_i = (|x_i| - a)_+ \text{sgn}(x_i). \quad (8)$$

The detailed procedure of our solution is described in Algorithm 1. In our implementation, we set $\lambda = 0.02$. Thanks to its computational simplicity and good convergence performance, FISTA is very efficient for sparse palmprint representation. Compared with other methods, in our palmprint verification experiments the FISTA method [2] is about 13 times faster than l_1 magic [3] and 3 times faster than FOCUSS [7] yet achieving the same verification accuracy.

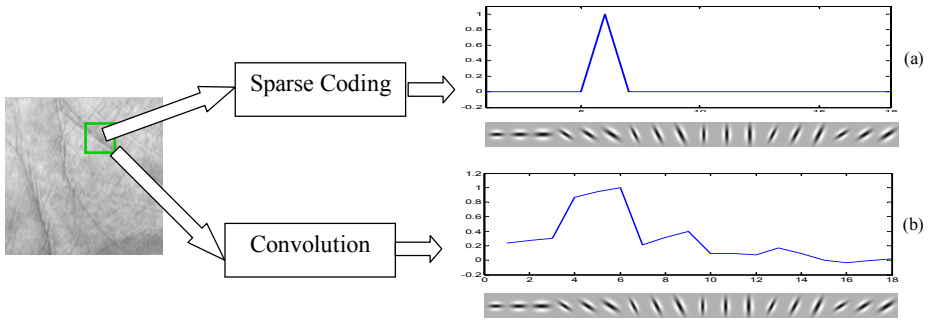

Figure 2. Representation of a palmprint image patch using: (a) sparse coding, and (b) regular convolution.

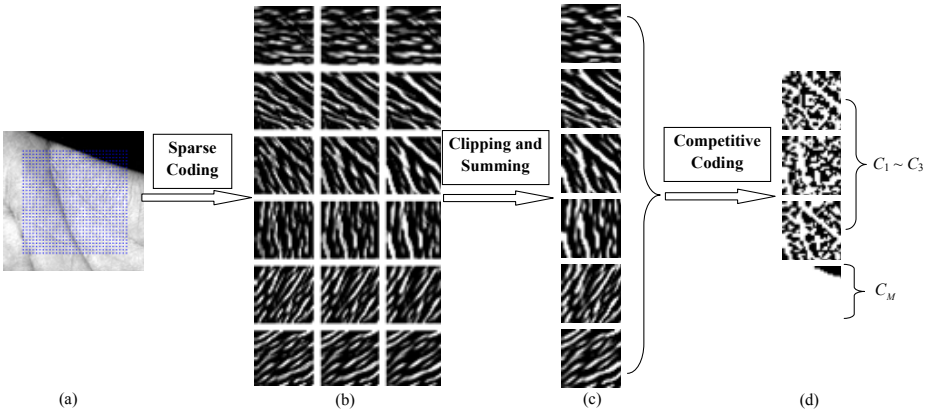

Figure 3. Schematic diagram of the feature extraction procedure of the SMCC. First, sparse coding is used to calculate the coefficients of 18 filters in the dictionary that correspond to 3 scales and 6 orientations. This is done for all patches centered at the blue pixels in the palmprint image (a). The sparse codes of all the patches are arranged into 18 coefficient images in (b), where the images in the same row are in different scales while those in the same column are in different orientations. Second, negative coefficients are replaced with zeros and the coefficient images in the same row are summed to construct a combined receptive field (c). Finally, the competitive code [17] (d) is generated for binary palmprint representation, which includes three binary feature matrices and one palmprint mask.

Algorithm 1 Sparse Palmprint Representation via FISTA

Input: $\boldsymbol{D}$, $\boldsymbol{x}_{ij}$, λ, L, $\boldsymbol{\alpha}_0 \leftarrow \boldsymbol{D}^T\boldsymbol{x}_{ij}$, $\boldsymbol{y}_1 \leftarrow \boldsymbol{\alpha}_0$, $t_1 \leftarrow 1$, $k \leftarrow 1$

Output: $\boldsymbol{\alpha}_{ij} \leftarrow \boldsymbol{a}_k$

1: **while** not converged **do**
2: $\quad \boldsymbol{a}_k \leftarrow \mathcal{T}_{\lambda/L}\left(\boldsymbol{y}_k - \nabla f(\boldsymbol{y}_k)/L\right)$
3: $\quad t_{k+1} \leftarrow \frac{1+\sqrt{1+4t_k^2}}{2}$
4: $\quad \boldsymbol{y}_{k+1} \leftarrow \boldsymbol{a}_k + \frac{t_k - 1}{t_{k+1}}(\boldsymbol{a}_k - \boldsymbol{a}_{k-1})$
5: $\quad k \leftarrow k + 1$
6: **end while**

Finally, we provide an example to compare the results of the convolution method and the proposed sparse representation method. As shown in Figure 2, the representation of the patch image using the proposed method is much sparser than that of using the regular convolution.

3. The SMCC for Palmprint Verification

The filter coefficients obtained by sparse coding in Section 2 are only a general sparse representation of the multiscale orientation field of palmprint. In this section, we further investigate the compact representation of the multiscale orientation field, and also discuss the matching method for effective and efficient palmprint verification.

3.1. Coding

To derive a compact and effective representation of the sparse multiscale orientation field, we proposed a sparse multiscale competitive code method by generalizing the competitive code [17].

Figure 3 shows the feature extraction procedure of the SMCC method, which involves three major steps: sparse coding, clipping and summing, and competitive coding. The details of SMCC are described as follows:

- Sparse coding: Given a 128×128 palmprint image, we uniformly sample 32×32 patches, where the horizontal and vertical distances between two adjacent centers

of patches are three pixels. For each patch $\boldsymbol{x}_{ij}$ centered at pixel (i, j), we use the FISTA algorithm [2] to calculate the filter coefficients $\boldsymbol{\alpha}_{ij}$. So we obtain 18 coefficient images that correspond to the 3 scales and 6 orientations.

- Clipping and summing: The response to the correct orientation and scale should be positive. So after sparse coding, all the negative values of the sparse multiscale orientation field are clipped to zeros. Next, we sum the clipped coefficient images in the same orientation to construct a combined receptive field $\boldsymbol{\omega}_{ij}$ of the palmprint image.

- Competitive coding: The competitive code [17] encodes the dominant orientation of the receptive field into a binary representation and utilizes the angular distance to measure the dissimilarity. Let $\boldsymbol{\omega}_{ij}(\theta)$ denote the combined receptive field $\boldsymbol{\omega}_{ij}$ in the θth direction and $W(i, j)$ be the winning index of the orientation in the patch $\boldsymbol{x}_{ij}$:

$$W(i,j) = \arg\min_{\theta} \boldsymbol{\omega}_{ij}(\theta). \tag{9}$$

If the number t of orientations is even, a coding rule [17],

$$C(i,j,k) = \begin{cases} 1, & k \le W(i,j) \le k + t/2, \\ 0, & \text{otherwise}, \end{cases} \quad k = 1, ..., t/2, \tag{10}$$

is defined to compactly encode each winning index into $t/2$ bits for efficient matching of palmprint features. Besides, as it is possible that not all pixels in the palmprint image are inside the palm we further add one mask bit to indicate the palmprint pixels. So the final SMCC code of a palmprint image consists of $t/2 + 1$ binary matrices $\boldsymbol{C} = \{C_1, C_2, ..., C_{t/2}, C_M\}$, where $C_k = (C(i, j, k))$ and C_M is the palmprint mask.

3.2. Matching

We use the angular distance proposed in [17] to measure the dissimilarity between SMCC codes. Given two SMCC codes $\boldsymbol{P} = \{P_1, P_2, ..., P_{t/2}, P_M\}$ and $\boldsymbol{Q} = \{Q_1, Q_2, ..., Q_{t/2}, Q_M\}$, their angular distance is defined as

$$d_A = \frac{\sum_{x,y=1}^{N} \sum_{i=1}^{t/2} \{P_M(x,y) \wedge Q_M(x,y)\} \wedge \{P_i(x,y) \otimes Q_i(x,y)\}}{\frac{t}{2} \sum_{x,y=1}^{N} (P_M(x,y) \wedge Q_M(x,y))} \tag{11}$$

where $\otimes$ is the bitwise exclusive OR (XOR) operator and $\wedge$ is the bitwise AND operator.

To reduce the adverse effect of image translation and rotation, we further divide each binary matrix in the SMCC code into four submatrices. For every combination of translation and rotation, for each submatrix we record the partial angular distance between the sample SMCC code and the query SMCC code, i.e., the summations in (11) are only for pixels (x, y) that are within the overlapping region of two corresponding submatrices, and then use the average of the four partial angular distances as the mismatching score under the combination of translation and rotation. The minimum mismatching score is recorded as the final mismatching score. Then the palmprint in the database having the smallest mismatching score with the query palmprint is considered as the best match.

4. Experimental Results

In this section, we use two popular palmprint databases, PolyU and CASIA, to evaluate the verification performance of the SMCC method. Several state-of-the-art mono-scale methods, such as competitive code [17], ordinal code [26], and robust line orientation code (RLOC) [14], are also implemented to compare with the SMCC. Besides, we also cite the performance of several recently developed methods [5, 8, 11] and several multiscale approaches [10, 31] reported in literature.

4.1. Experimental Results on the PolyU Palmprint Database

The PolyU palmprint database (version 2)[1] contains 7,752 palmprint images from 193 individuals, where 131 individuals are male. The age distribution is: subjects aged less than 30 accounts for 86 percent, those older than 50 is about 3 percent, and those between 30 and 50 is about 11 percent. The samples of each individual were collected in two sessions, where the average interval between the first and the second sessions was around two months. In each session, each individual was asked to provide about 10 images of each palm. Then each original palmprint image was cropped to a size of 128×128.

In our verification experiments, each palmprint image is compared with all the other palmprint images in the database. The total number of matches is 30,042,876, which includes 74,068 genuine and 29,968,808 imposter matches. We use the receiver operating characteristic (ROC) curve, equal error rate (EER), and the false reject rate (FRR) at specific false accept rate to evaluate the verification performance of the palmprint recognition methods.

Figure 4 shows the ROC curves of SMCC and several state-of-the-art palmprint recognition methods. One can see that for any given FAR, the genuine accept rate (GAR) of SMCC is always higher than those of other methods.

[1] http://www.comp.polyu.edu.hk/ biometrics/

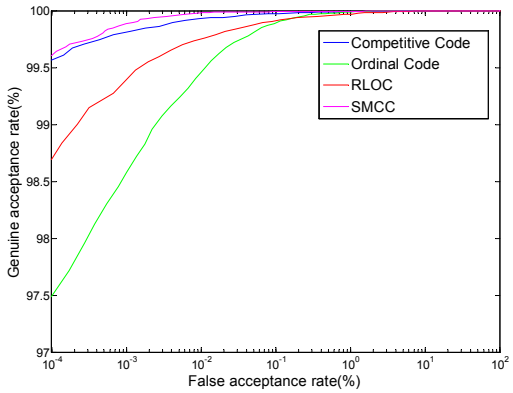

Figure 4. The ROC curves of different palmprint verification methods on the PolyU palmprint database (version 2).

Table 1. Comparison of error rates of different palmprint verification methods on the PolyU palmprint database.

Method	EER (%)	$GAR_{-2}(\%)$	$GAR_{-3}(\%)$
SMCC	**0.014**	**99.98**	**99.89**
CompCode [17]	0.038	99.93	99.81
RLOC [14]	0.091	99.75	99.40
OrdiCode [26]	0.104	99.46	98.58

Table 1 lists the values of GAR at several typical FAR values (GAR_{-2} and GAR_{-3}) and the values of EER of SMCC, competitive code (CompCode) [17], ordinal code (OrdiCode) [26], and RLOC [14]. We use GAR_{-2} (GAR_{-3}) to denote the genuine acceptance rate when FAR = $10^{-2}\%$ ($10^{-3}\%$). Compared with these methods, the SMCC achieves the lowest EER value and the highest GAR_{-2} and GAR_{-3} values.

It is important to evaluate whether the improvement of error rates of the SMCC is statistically significant. Following [9], we discuss the statistical significance in performance difference of different algorithms. Let $\alpha = 0.05$ be the confidence interval, and e and $\hat{e}$ be the error rate of a classifier C and the estimated error rate using a test set with finite samples, respectively. Assume that recognition errors are Bernoulli trials. Given a typical value $\beta = 0.2$, Guyon *et al*. [9] proposed a simple equation to determine the number of trials $N \approx 100/e$ to achieve $1 - \alpha$ confidence in that the error rate estimate is within the range $|e - \hat{e}| \leq \beta e$. In our experiments, the number of genuine and imposter comparisons are 74,068 and 29,968,808, respectively. Thus the statistical significance could be guaranteed with an empirical error rate down to $0.33 \times 10^{-3}\%$. From Table 1, the EER, the FRR_{-2} (= $1 - GAR_{-2}$), and FRR_{-3} (= $1 - GAR_{-3}$) values of the SMCC are both higher than $0.33 \times 10^{-3}\%$ and 20% lower than those of other methods. So we can conclude that the improvement of error rates of the SMCC on the PolyU palmprint database is statistically significant.

Table 2. Feature extraction (FeaExt) and mathcing time (ms) and template sizes (bytes) of different palmprint verification methods.

Method	FeaExt	Matching	Template Size
SMCC	330	0.05	384
CompCode [17]	70	0.04	384
RLOC [14]	2.2	0.85	1024
OrdiCode [26]	58	0.04	384

Note: The experimental environment is: Windows XP Professional, Pentium 4 2.66GHz, 512M RAM, VC 6.0.

Table 3. Comparison of the SMCC with three recent methods in terms of EER (%) and template size (bytes) on the PolyU database.

Method	EER (%)	Template Size (bytes)
SMCC	0.014	384
GMBoostLDA [5]	0.20	1400*
RAC [11]	0.01	184320*
BOCV [8]	0.019	768

Note: The template sizes of these two methods are our estimate, based on the dimension of features and if the features are stored in 8-byte numbers.

Table 4. Comparison of error rate (%) and template size (bytes) of different multiscale palmprint verification methods on the PolyU database.

Method	EER (%)	Template Size (bytes)
SMCC	0.014	384
MCC [31]	0.023	480
HAS [10]	0.02	672*

Note: The number "672" could not be directly obtained from literature. It is our estimate.

Table 2 lists the computational time of SMCC, competitive code (CompCode) [17], ordinal code (OrdiCode) [26], and RLOC [14]. The feature extraction time of SMCC is 330ms, which is higher than the other three methods but is still sufficient for practical use. The matching time of SMCC is 0.05ms, which is much lower than that of RLOC [14] and is only slightly higher than that of competitive code [17] and ordinal code [26].

We look into three recent methods: Gabor magnitude feature (GMBoostLDA) [5], regional appearance correlation (RAC) [11], and binary orientation co-occurrence vector (BOCV) [8], and collect the statistics of their performance on the PolyU database in Table 3, where the EER of GMBoostLDA is estimated from the ROC curve in [5]. We can see that the SMCC is superior to GMBoostLDA [5] and BOCV [8] in terms of EER and template size. Compared with SMCC, RAC [11] achieves a slightly lower EER than SMCC, but has a much larger template size. So by taking both accuracy and template size into account, the SMCC is very promising.

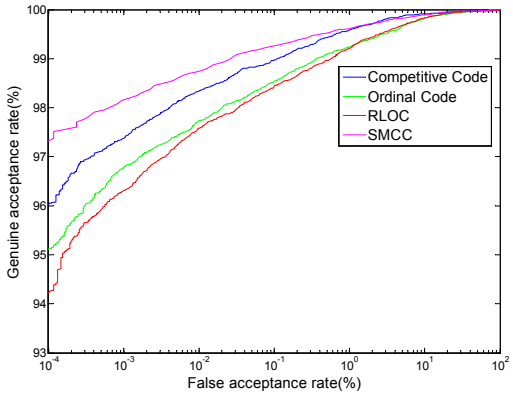

Figure 5. The ROC curves of different palmprint verification methods on the CASIA palmprint database.

Table 5. Comparison of error rates of different palmprint verification methods on the CASIA palmprint database.

Method	EER (%)	$GAR_{-2}(\%)$	$GAR_{-3}(\%)$
SMCC	**0.48**	**98.74**	**98.16**
CompCode [17]	0.55	98.34	97.60
RLOC [14]	0.81	97.60	96.31
OrdiCode [26]	0.84	97.74	96.80

We also compare the EER value of SMCC with those of two multiscale palmprint recognition methods, the multiscale competitive code (MCC) [31] and hierarchical appearance statistics (HAS) [10] and show the results in Table 4. It should be noted that, for the MCC and HAS we use the EER values reported in [31] and [10], which are also obtained using the PolyU palmprint database. One can see that the SMCC achieves the lowest EER value.

Finally, we compare the sizes of templates used by different methods. Table 2 and Table 3 list the template sizes of the SMCC and several mono-scale methods, while Table 4 presents the template sizes of multiscale methods. The template size of the SMCC is smaller than those of multiscale methods. Although the template sizes of the competitive code [17] and the ordinal code [26] are equal to that of the SMCC, they only record mono-scale information, while the SMCC records multiscale information. Thus the SMCC is a more compact and effective multiscale palmprint representation method.

4.2. Experimental Results on the CASIA Palmprint Database

The CASIA palmprint database[2] contains 5,239 palmprint images from 301 individuals. To the best of our knowledge, this database is the largest publicly available database in terms of the number of subjects. The samples were collected in one session only. The subject was asked to provide about 8 palmprint images of his / her left and right palms. In our experiments, we found that individual "101" is the same as individual "19". So we merged the images in these two folders into one folder. Thus, the number of individuals is 300 and the number of palms is 600.[3]

For the CASIA palmprint database, the total number of matches is 13,710,466, which includes 20,574 genuine and 13,689,892 imposter matches. Figure 5 shows the ROC curves of the SMCC, CompCode [17], OrdiCode [26], and RLOC [14]. From Figure 5, it is observed that for any given FAR value, the genuine accept rate of the SMCC is always higher than those of the other three palmprint recognition methods.

Table 5 lists the GAR values at several typical FAR values (GAR_{-2} and (GAR_{-3}) and the EER values of the SMCC, CompeCode [17], OrdiCode [26], and RLOC [14]. The SMCC again achieves the lowest EER value and the highest GAR_{-2} and GAR_{-3} values. Using the same method for the experiments on the PolyU database, we can also testify to the statistical significance in the performance improvement of error rates of the SMCC on the CASIA palmprint database. These results indicate that the SMCC is superior to the existing palmprint verification methods.

Comparing Table 1 with Table 5, one can observe that the error rates obtained using the CASIA database are much higher than those obtained using the PolyU palmprint database. This may be caused by three reasons. First, the images in CASIA were captured using web cameras, and thus the image quality was not as good as that from PolyU. Second, there were no pegs to restrict postures and positions of palms during the data collection of CASIA, which brought a large degree of freedom. Finally, the number of palms in CASIA is larger than that in PolyU, which may make palmprint verification more difficult.

5. Conclusion

In this paper, we propose a method called the sparse multiscale competitive code for palmprint verification. The SMCC first uses the sparse coding method to obtain a sparse representation of the multiscale orientation field, and then extends the competitive code [17] for compact and effective multiscale palmprint representation. Compared with the existing mono-scale and multiscale palmprint recognition methods, the SMCC achieves better verification performance, yet requires a smaller template size than the multiscale palmprint recognition methods.

Compact and effective local feature representation is an important topic in computer vision. The SMCC is insensitive to illumination and scaling factors, and thus is expected

[2] http://www.csbr.ia.ac.cn

[3] Two poor quality images were excluded from our experiments because they lack necessary fiducial points for preprocessing.

to be effective as a potential texture descriptor for several other biometric recognition and vision tasks.

Acknowledgment

This work is partially supported by Microsoft Research Asia, the National Natural Science Foundation of China under Contract Nos. 60902099 and 60872099, and the Education Ministry Young Teacher Foundation of China under No. 200802131025.

References

[1] M. Aharon, M. Elad, and A. M. Bruckstein. The k-svd: an algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Processing*, 54(11):4311–4322, 2006.

[2] A. Beck and M. Teboulle. A fast iterative shrinkage- thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[3] E. Cands and J. Romberg. l_1-magic: Recovery of sparse signals via convex programming. Technical report, California Institute of Technology, 2005.

[4] E. Cands, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Commun. Pure Appl. Math.*, 59(8):1207C1223, 2005.

[5] R. Chu, Z. Lei, Y. Han, R. He, and S. Z. Li. Learning gabor magnitude features for palmprint recognition. In *ACCV 2007*, pages 22–31, 2007.

[6] R. Garg and R. Khandekar. Gradient descent with sparsification: an iterative algorithm for sparse recovery with restricted isometry property. In *International Conference on Machine Learning 2009*, 2009.

[7] I. F. Gorodnitsky and B. D. Rao. Sparse signal reconstruction from limited data using focuss: a reweighted minimum norm algorithm. *IEEE Trans. Signal Processing*, 45(3):600–616, 1997.

[8] Z. Guo, D. Zhang, L. Zhang, and W. Zuo. Palmprint verification using binary orientation co-occurrence vector. *Pattern Recognition Letters*, 30(13):1219–1227, 2009.

[9] I. Guyon, J. Makhoul, R. Schwartz, and V. Vapnik. What size test set gives good error rate estimates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(1):52–64, 1997.

[10] Y. Han, Z. Sun, and T. Tan. Combine hierarchical appearance statistics for accurate palmprint recognition. In *ICPR 2008*, 2008.

[11] Y. Han, Z. Sun, T. Tan, and Y. Hao. Palmprint recognition based on regional rank correlation of directional features. In *International Conference on Biometrics 2009*, pages 587–596, 2009.

[12] Y. Han, T. Tan, and Z. Sun. Palmprint recognition based on directional features and graph matching. In *International Conference on Biometrics 2007*, pages 1164–1173, 2007.

[13] A. K. Jain and J. Feng. Latent palmprint matching. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(5):1032–1047, 2009.

[14] W. Jia, D. S. Huang, and D. Zhang. Palmprint verification based on robust line orientation code. *Pattern Recognition*, 41(5):1504–1513, 2008.

[15] S. Kim, S. Koh, M. Lustig, S. Boyd, and D. Gorinevsky. An interior-point method for large scale l_1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1(4):606–617, 2007.

[16] A. Kong, D. Zhang, and M. Kamel. Palmprint identification using feature-level fusion. *Pattern Recognition*, 39(3):478–487, 2006.

[17] A. W. K. Kong and D. Zhang. Competitive coding scheme for palmprint verification. In *ICPR 2004*, pages 520–523, 2004.

[18] W. K. Kong, D. Zhang, and W. X. Li. Palmprint feature extraction using 2-d gabor filters. *Pattern Recognition*, 36(10):2339–2347, 2003.

[19] K. Kreutz-Delgado, J. F. Murray, B. D. Rao, K. Engan, T. W. Lee, and T. J. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural Computation*, 15(2):349–396, 2003.

[20] A. Kumar and D. Zhang. Personal authentication using multiple palmprint representation. *Pattern Recognition*, 38(10):1695–1704, 2005.

[21] S. I. Lee, H. Lee, P. Abbeel, and A. Ng. Efficient l_1 regularized logistic regression. In *AAAI-06*, 2006.

[22] W. Li, J. You, and D. Zhang. Texture-based palmprint retrieval using a layered search scheme for personal identification. *IEEE Trans. Multimedia*, 7(5):891–898, 2005.

[23] G. Lu, D. Zhang, and K. Wang. Palmprint recognition using eigenpalms features. *Pattern Recognition Letters*, 24(9):1463–1467, 2003.

[24] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. In *Neural Information Processing Systems 2008*, 2008.

[25] R. K. Rowe, U. Uludag, M. Demirkus, S. Parthasaradhi, and A. K. Jain. A spectral whole-hand biometric authentication system. In *2007 Biometric Symposium*, 2007.

[26] Z. Sun, T. Tan, Y. Wang, and S. Z. Li. Ordinal palmprint representation for personal identification. In *CVPR 2005*, pages 279–284, 2005.

[27] J. You, W. K. Kong, D. Zhang, and K. H. Cheung. On hierarchical palmprint coding with multiple features for personal identification in large databases. *IEEE Trans. Circuits and Systems for Video Technology*, 14(2):234–243, 2004.

[28] F. Yue, W. Zuo, K. Wang, and D. Zhang. A performance evaluation of filter design and coding schemes for palmprint recognition. In *ICPR 2008*, 2008.

[29] D. Zhang, W. K. Kong, J. You, and M. Wong. Online palmprint identification. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1041–1050, 2003.

[30] D. Zhang, G. Lu, W. Li, L. Zhang, and N. Luo. Palmprint recognition using 3-d information. *IEEE Trans. Systems, Man, and Cybernetics-Part C*, 39(5):505–519, 2009.

[31] W. Zuo, F. Yue, K. Wang, and D. Zhang. Multiscale competitive code for efficient palmprint recognition. In *ICPR 2008*, 2008.