Image and Vision Computing 30 (2012) 1043–1051

Contents lists available at SciVerse ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Encoding local image patterns using Riesz transforms: With applications to palmprint and finger-knuckle-print recognition☆

Lin Zhang, Hongyu Li *

School of Software Engineering, Tongji University, Shanghai, China

ARTICLE INFO

Article history:
Received 2 January 2012
Received in revised form 5 June 2012
Accepted 21 September 2012

Keywords:
Biometrics
Riesz transforms
Palmprint recognition
Finger-knuckle-print recognition

ABSTRACT

Biometrics authentication is an effective method for automatically recognizing a person's identity with high confidence. It is well recognized that in biometric systems feature extraction and representation are key considerations. Among various feature extraction and representation schemes, coding-based methods are most attractive because they have the merits of high accuracy, robustness, compactness and high matching speed, and thus they have been adopted in many different kinds of biometric systems, such as iris, palmprint, and finger-knuckle-print based ones. However, how to devise a good coding scheme is still an open issue. Recent studies in image processing and applied mathematics have shown that local image features can be well extracted with Riesz transforms in a unified framework. Thus, in this paper we propose to utilize Riesz transforms to encode the local patterns of biometric images. Specifically, two Riesz transforms based coding schemes, namely RCode1 and RCode2, are proposed. They both use 3-bits to represent each code and employ the normalized Hamming distance for matching. RCode1 and RCode2 are thoroughly evaluated and compared with the other 3-bit coding methods on a palmprint database and a finger-knuckle-print database. Experiments show that the proposed methods, especially RCode2, could achieve quite similar verification accuracies with the state-of-the-art method (CompCode) while they need much less time at the feature extraction stage, which renders them better candidates for time critical applications.

1. Introduction

Recognizing the identity of a person with high confidence is a critical issue in various applications, such as e-banking, access control, passenger clearance, national ID card, etc. The need for reliable user authentication techniques has significantly increased in the wake of heightened concerns about security, and rapid advancement in networking, communication and mobility [12]. Biometrics based methods, which use unique physical or behavioral characteristics of human beings, are of broad interest and have great potentials because of their high accuracy and convenience to use in the modern e-world. In fact, researchers have exhaustively investigated a number of different biometric identifiers, including fingerprint, face, iris, palmprint, hand geometry, voice, gait, etc.

Among various kinds of biometric identifiers, hand-based biometrics has been attracting considerable attention over recent years because of their high user acceptance. Fingerprint [10,22], palmprint [1,7,11,13–16,20,28,31,39,42], hand geometry [3], hand vein [35], and finger-knuckle-print [17,18,24,38,40,44–46] have been proposed and investigated in the literature. In this paper, we lay our focus on palmprint recognition and finger-knuckle-print recognition. Palmprint refers to the skin patterns in the inner palm surface, comprising mainly two kinds of features: the palmar friction ridges (the ridge and valley structures like the fingerprint) and the palmar flexion creases (discontinuities in the epidermal ridge patterns) [11]. Different image resolutions are required to observe different palmprint features. For example, major creases can be observed at less than 100 dpi, while resolutions greater than 400 dpi are needed to observe thin creases and ridges [11,16]. Roughly speaking, high resolution palmprints are needed in forensic applications such as criminal detection, while low resolution palmprints are suitable for civil applications such as access control. In this paper, we deal with the latter case. Compared with palmprint, the finger-knuckle-print (FKP) is still in its infancy as a biometric identifier and has not attracted enough attention from researchers. FKP refers to the inherent skin patterns of the outer surface around the phalangeal joint of one's finger. Pioneering works from different research groups have demonstrated that FKP is highly unique for different individuals and thus can serve as a biometric identifier [17,18,24,38,40,44–46].

Generally speaking, as shown in Fig. 1, a typical palmprint or FKP recognition system consists of the following three stages: preprocessing and ROI (region of interest) extraction, feature extraction, and feature matching. Preprocessing and ROI extraction algorithms for palmprint and FKP can be found in [42] and [46], respectively. Fig. 2a and b shows a sample palmprint image taken from [27] and its corresponding ROI. Fig. 2c and d shows a sample FKP image taken from [26] and its

☆ This paper has been recommended for acceptance by Mark Nixon.
* Corresponding author.
E-mail address: hyli@tongji.edu.cn (H. Li).

0262-8856/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.imavis.2012.09.003

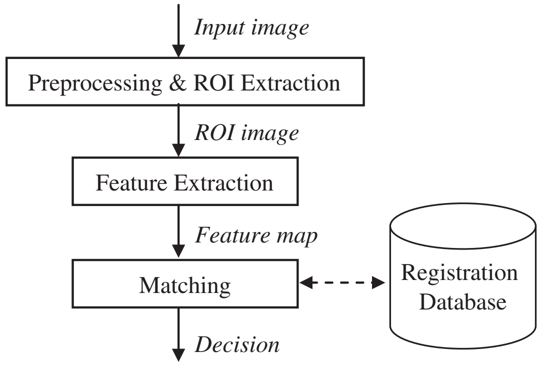

Fig. 1. Architecture of a typical palmprint or FKP recognition system.

corresponding ROI. In this paper, our efforts are focused on the second stage, i.e., finding a good way to represent the palmprint or the FKP information using a compact set of features. As in any pattern classification task, such a feature extraction step plays a vital role in a palmprint or FKP based personal authentication system. Palmprint images and FKP images share some similar traits. For example, they both hold abundant line features. So we will treat them as a class of images and expect to derive some common feature extraction and representation methods that can be applied to them both. In fact, most of the existing FKP recognition methods are adapted from palmprint recognition methods previously developed.

In the past decade or so, an enormous volume of literature has been devoted to investigate various feature extraction methods for the palmprint recognition and according to Kong et al. [16] these methods can be mainly classified into four groups: line-based, subspace-based, statistic-based, and coding-based. Several of them have also been adapted for FKP recognition. In line-based approaches, palm lines are extracted using newly developed or existing line detection operators. Then, these lines are matched directly or represented in other formats for matching. Two representative approaches belonging to this category were proposed in [13,20]. Subspace-based methods, such as the principal component analysis (PCA), the linear discriminant analysis (LDA), and the independent component analysis (ICA), are also explored for the palmprint recognition [1,28,39]. In [17], Kumar and Ravikanth combined PCA, LDA, and ICA to perform FKP recognition. In [40], Yang et al. proposed a multi-manifold discriminant analysis (MMDA) method for FKP matching based on graph embedded learning under the Fisher discriminant analysis framework. These methods attempt to find a set of basis images from a training set and represent any probe image as a linear combination of these basis images. Usually however, this kind of methods cannot achieve very high verification accuracies. Local or global statistical measures, such as Hu moments, Zernike moments, centers of gravity, standard deviation, variance, and histograms of local binary patterns, are also exploited as palmprint features [16]. Inspired by the great success of IrisCode invented by Daugman [2], researchers also propose coding-based matching schemes for palmprint recognition. In a typical coding-based method, each field of the code map is assigned a bit-wised code, based on the quantization of the image's responses to a set of filters. Several coding-based algorithms have been proposed for palmprint identification. PalmCode uses a single Gabor filter to extract the local phase information of palmprint [42]. Its computational architecture is the same as IrisCode. In [15], Kong and Zhang proposed the competitive code (CompCode) scheme, which encodes the local orientation field of a palmprint using symmetric Gabor filters along six orientations. Later, CompCode was extended for FKP recognition in [44]. In [14], Jia et al. proposed another coding method to extract the local orientation information of palmprints, namely robust line orientation code (RLOC), which is based on the modified finite Radon transform. RLOC was also adapted for FKP recognition in [18]. In [31], Sun et al. used differences between Gaussians to extract the local ordinal measures from palmprints. For FKP recognition, there are also some other methods that are not easy to be categorized. In [24], Morales et al. used a real Gabor filter to enhance the FKP image and then used the scale invariant feature transform (SIFT) [21] to extract features; two

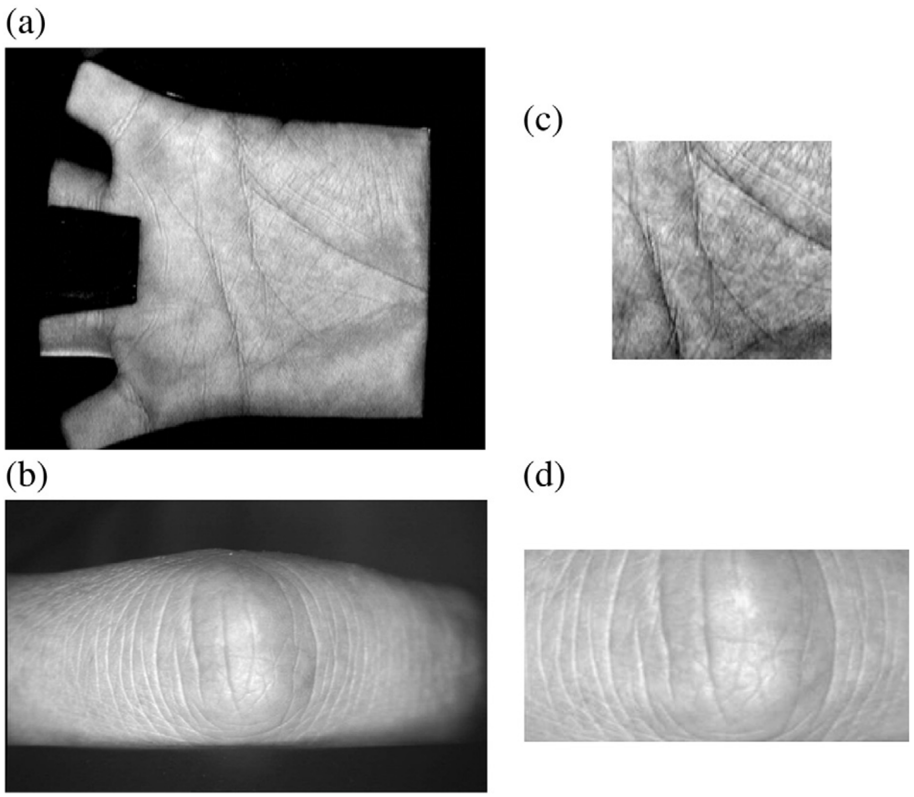

Fig. 2. (a) is a sample palmprint image taken from [27]; (b) is the ROI extracted from (a); (c) is a sample FKP image taken from [26]; and (d) is the ROI extracted from (c).

FKP image's dissimilarity are computed based on their SIFT feature sets. In the local–global information combination feature extraction scheme [45], the local orientation extracted by Gabor filters is taken as the local feature while the Fourier coefficients are taken as a global feature.

Compared with the other methods, coding-based approaches have many advantages. Generally speaking, they can usually achieve a good balance on various performance metrics, such as the verification accuracy, the robustness to illumination variation, the feature size, the feature extraction speed, and the feature matching speed. Daugman's IrisCode [2] can serve as an excellent successful example. He has demonstrated that the real-time brute-force identification in large databases is possible with coding based features and the associated bit-wise Hamming distance based matching. Taking these factors into consideration, in this paper we lay our focus on devising novel efficient and effective coding-based matching methods for palmprint/FKP recognition. Undoubtedly, the core of a coding-based method is how to devise a "perfect" coding scheme. To this end, the following factors should be carefully considered: the computational complexity, the compactness (number of bits for each code), the robustness to the illumination changes, and the distinctiveness. Essentially, the local orientation (e.g., CompCode), the local phase (e.g., PalmCode), or some other the local variations (e.g., OrdinalCode) will be reflected in a coding scheme. Based on recent studies in the signal/image processing community, the local image information can be well characterized in a unified theoretic framework, namely Riesz transform [29], which actually is a vector-valued extension of the Hilbert transform. Recently, Riesz transforms have attracted much attention in the signal/image processing community [4,5,9,33,34,36,37,41]. Felsberg and Sommer are the first to bring the Reisz transform to the signal/image processing community [4]. In their work, they proposed the monogenic signal based on the 1st-order Riesz transform. The monogenic signal can be regarded as a 2D extension to the classical analytic signal. Using the monogenic signal, the local phase and the local orientation of the intrinsic 1D signal can be extracted and represented in a compact and isotropic way. The monogenic signal has already been used in some image-processing related applications, such as the local structure analysis [5,41], the stereo motion estimation [32], the image registration [23], and the texture classification [43], etc. However, from the theoretical view, 1st-order Riesz transform based monogenic signal is designed only for intrinsic 1D signal. In order to characterize the intrinsic 2D local structures, higher order Riesz transforms are required. In [37], Wietzke and Sommer used up to 3rd-order Riesz transforms to construct the signal multi-vector which allows modeling any local image structures, such as lines, edges, corners, and junctions, in a scale space with one unified framework. Since by using Riesz transforms local image patterns can be well represented in a unified framework, in this paper we propose to quantify the image's responses to the Riesz transforms to devise new coding schemes. Specifically, two different Riesz transforms based coding schemes are proposed, namely RCode1 and RCode2. They both use 3-bits to represent each code and resort to the normalized Hamming distance for matching. We conduct comparative experimental studies of different coding schemes on the PolyU palmprint database [27] and the PolyU finger-knuckle-print database [26]. Experimental results show that the two newly proposed coding methods, especially RCode2, could achieve high verification accuracies for both the palmprint recognition and the FKP recognition. RCode2 could get similar verification accuracy with the state-of-the-art 3-bit coding method CompCode [15], while it has the advantage of computational simplicity for the feature extraction, making it a better choice for time-critical applications.

The remainder of this paper is organized as follows. Section 2 introduces three state-of-the-art coding based feature extraction methods used for palmprint and finger-knuckle-print recognition. Section 3 presents a short introduction for Riesz transforms. Section 4 presents our new coding schemes based on Riesz transforms. Section 5 reports the experimental results. Finally, Section 6 concludes the paper.

2. Existing coding methods for palmprint and FKP recognition

In this section, three state-of-the-art coding-based feature extraction and matching schemes used for palmprint recognition will be briefly reviewed. Two of them, CompCode [15] and RLOC [14], have also been adapted for FKP recognition [44,18].

2.1. Competitive code (CompCode)

The neurophysiology-based Gabor filter proposed by Lee [19] is used in CompCode. It is defined as

$$G(x,y) = \frac{\omega}{\sqrt{2\pi}\kappa} e^{-\left(\omega^2/8\kappa^2\right)\left(4x'^2+y'^2\right)} \left(e^{i\omega x'} - e^{-\kappa^2/2}\right) \tag{1}$$

where $x' = x\cos\theta + y\sin\theta$, $y' = -x\sin\theta + y\cos\theta$, ω is the radial frequency in radians per unit length and θ is the orientation of the Gabor filter in radians. κ is defined by $\kappa = \sqrt{2\ln 2}\left(\left(2^\delta+1\right)/\left(2^\delta-1\right)\right)$, where δ is the half-amplitude bandwidth of the frequency response. ω can be determined by $\omega = \kappa/\sigma$, where σ is the standard deviation of the Gaussian envelop.

Denote by G_R the real part of the Gabor filter G. With six G_Rs sharing the same parameters, except the parameter of orientation, as shown in Fig. 3, the local orientation information of the image I at the position (x, y) can be extracted and coded. Mathematically, this competitive coding process can be expressed as

$$CompCode(x,y) = \arg\min_j \left\{ I(x,y) * G_R\left(x,y,\theta_j\right) \right\} \tag{2}$$

where $*$ stands for the convolution operation and $\theta_j = j\pi/6, j = \{0,\ldots,5\}$. Obviously, each $CompCode(x, y)$ is actually an integer within 0–5, representing the local dominant orientation index. For efficient representation and bitwise matching, dominant orientation indices {0, 1, 2, 3, 4, 5} are encoded with three bits as {000, 001, 011, 111, 110, 100}. It can be seen that six convolutions with the image are needed to extract its CompCode.

2.2. Robust line orientation code (RLOC)

Robust line orientation code (RLOC) proposed by Jia et al. [14] is another effective method to encode the local orientation information of an image. It assumes that lines in the examined image are negative lines. For palmprint line orientation estimation, Jia et al. [14] devised six binary line templates, T_0~T_5, with line width $w = 4$ pixels, as shown in Fig. 4. These six line templates are of the orientations 0, π/6, π/3, π/2, 2π/3, and 5π/6, respectively. In Fig. 4, "gray" fields are of value 1 while "white" fields are of value 0. By using the "winner-take-all" rule, RLOC is defined as

$$RLOC(x,y) = \arg\min_j \left\{ I(x,y) * T_j(x,y) \right\}, j \in \{0,\ldots,5\}. \tag{3}$$

Then, each $RLOC(x, y)$ is coded with 3-bits using the same scheme as CompCode.

2.3. Ordinal code

In OrdinalCode [31], a 2D Gaussian filter is used to calculate the weighted average intensity of a line-like region. It can be expressed as

$$Gau(x,y,\theta) = \exp\left(-\frac{(x\cos\theta + y\sin\theta)^2}{\sigma_x^2} - \frac{(-x\sin\theta + y\cos\theta)^2}{\sigma_y^2} \right) \tag{4}$$

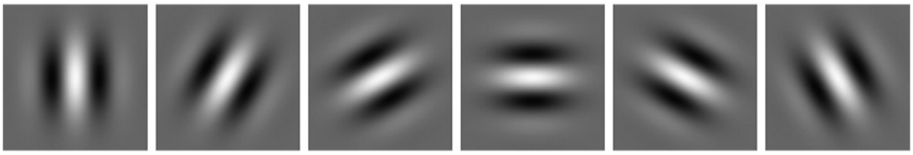

Fig. 3. Real parts of six Gabor filters with different orientations.

where θ denotes the dominant orientation of the 2D Gaussian filter, and σ_x and σ_y are the two standard deviations of the 2D Gaussian filter.

Based on the 2D Gaussian filter, the ordinal filter, which compares two orthogonal line-like image regions, is designed as

$$OF(x,y,\theta) = Gau(x,y,\theta) - Gau(x,y,\theta+\pi/2). \tag{5}$$

For each local region in the examined image, three ordinal filters $OF(0)$, $OF(\pi/6)$, and $OF(\pi/3)$, as shown in Fig. 5, are applied to it to obtain a 3-bit ordinal code based on the sign of the filtering results.

2.4. Matching

As stated, given an input image, three coding schemes CompCode, RLOC, and OrdinalCode will have similar outputs in form, i.e., code maps consisting of three bit-planes, although the filters and the coding strategies adopted are different. Therefore, they can use a universal matching scheme, namely the normalized Hamming distance, to measure the dissimilarity of two given code maps. Suppose that P and Q are two code maps extracted by any coding scheme described above. The normalized Hamming distance between P and Q is defined as

$$d(P,Q) = \frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}\sum_{i=0}^{2}\left(P_i^b(x,y)\otimes Q_i^b(x,y)\right)}{3S} \tag{6}$$

where $P_i^b(Q_i^b)$ is the ith bit-plane of $P(Q)$, S is the area of the code map, and $\otimes$ represents the bitwise "exclusive OR" operation.

In practice, taking into account the possible translation in the extracted ROI sub-image with respect to the one extracted in the enrolment, multiple matches are performed by translating one set of features in horizontal and vertical directions. And the minimum of the resulting matching distances is considered to be the final matching distance. In such cases, S is the area of the overlapping parts between two code maps.

3. Fundamentals of Riesz transforms

In this section, we give a brief review of the Riesz transforms. The mathematical fundamentals are partly based on the authoritative book of Stein [29] and partly on the extensive works of the Cognitive Systems Group in Kiel University [4,5,36,37,41].

3.1. Hilbert transform and the analytic signal

For $f \in L^p(\mathbb{R})$, $1 \le p < \infty$, the Hilbert transform of f, Hf, is defined as

$$(Hf)(x) = \frac{1}{\pi} p.v.\int_{\mathbb{R}} \frac{f(t)}{x-t}dt \tag{7}$$

where $p.v.$ stands for the Cauchy principal value. Equally, H can be expressed by the convolution kernel in the spatial domain as $h(x) = 1/\pi x$. The Fourier transform of the kernel h is $\hat{h}(u) = -ju/|u|$, where $j^2 = -1$.

The Hilbert transform based analytic signal, first proposed by Denis Gabor [6], is a powerful tool for the 1D signal analysis. Given a 1D real signal $f(x)$, the corresponding analytic signal is defined as

$$f_a(x) = f(x) + jHf(x) = A(x)e^{j\phi(x)} \tag{8}$$

where $A(x) = \sqrt{(f(x))^2 + (Hf(x))^2}$ is the local amplitude while $\phi(x) = \arctan 2(Hf(x), f(x)) \in [0, 2\pi)$ is the instantaneous phase. The advantage of this representation is that it allows one to obtain the time-varying amplitude and phase of a 1D signal. After its birth, the analytic signal has been utilized in various applications involving some kind of amplitude or frequency modulation [8,25].

3.2. 1st-order Riesz transform and the monogenic signal

In the literature, there are many attempts reported to generalize the analytic signal to 2D and among them, the monogenic signal [4] proposed by Felsberg and Sommer is the most distinguished one. The monogenic signal is built upon the 1st-order Riesz transform which is a vector-valued extension of the Hilbert transform [29]. The convolution kernels of the Riesz transform in the nD spatial domain can be expressed as

$$R_j(\mathbf{y}) = c_n \frac{\mathbf{y}_j}{|\mathbf{y}|^{n+1}} \tag{9}$$

where $c_n = \Gamma[(n+1)/2]/\pi^{(n+1)/2}$, $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and $j = 1, 2, \ldots, n$. In 2D, which is the case of interest for image processing applications (in this case, $n=2$, $j=\{1, 2\}$, and $c_n = 1/2\pi$), the Riesz transform consists of two kernels expressed as

$$h_x(\mathbf{x}) = \frac{x}{2\pi|\mathbf{x}|^3}, h_y(\mathbf{x}) = \frac{y}{2\pi|\mathbf{x}|^3} \tag{10}$$

where $\mathbf{x} = (x, y) \in \mathbb{R}^2$. Fourier transforms of h_x and h_y are

$$H_u(\mathbf{u}) = -j\frac{u}{|\mathbf{u}|}, H_v(\mathbf{u}) = -j\frac{v}{|\mathbf{u}|} \tag{11}$$

where $\mathbf{u} = (u, v) \in \mathbb{R}^2$.

Given a 2D signal $f(\mathbf{x})$, its corresponding monogenic signal $f_M(\mathbf{x})$ is defined as the combination of the original signal itself and its two Riesz transforms

$$f_M(\mathbf{x}) = \left(f(\mathbf{x}), h_x\{f\}(\mathbf{x}), h_y\{f\}(\mathbf{x})\right) \tag{12}$$

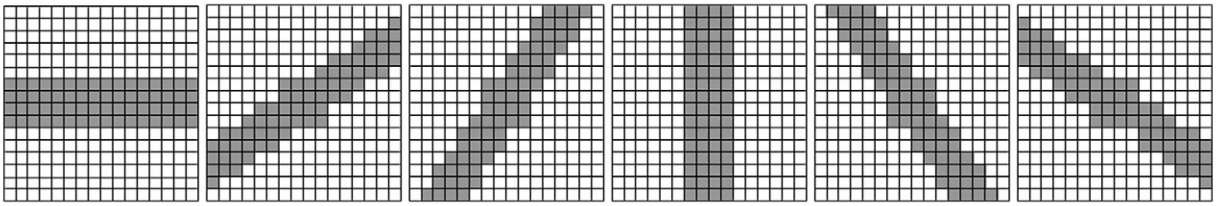

Fig. 4. Six binary line templates used in RLOC. "Gray" fields are of value 1 while "white" fields are of value 0.

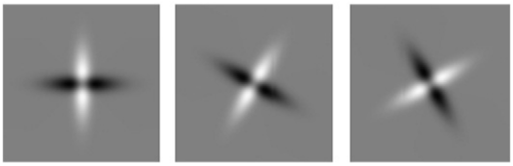

Fig. 5. Three ordinal filters $OF(0)$, $OF(\pi/6)$, and $OF(\pi/3)$ used in OrdinalCode.

where $h_x\{f\}$ means convolving f with h_x. As the Riesz transform with respect to the Hilbert transform, the monogenic signal is a multi-dimensional isotropic generalization of the 1D analytic signal.

To ease the following discussions, we introduce another concept here, the intrinsic dimension. The intrinsic dimension is the number of degrees of freedom necessary to describe a local image structure [36]. 2D image signals can be classified into local regions N of different intrinsic dimensions. For example, constant areas are of intrinsic dimension zero (i0D) while straight lines and edges are of intrinsic dimension one (i1D). Mathematically, such a classification can be expressed as

$$f \in \begin{cases} \mathrm{i0}D_N, f(\mathbf{x}_i) = f\left(\mathbf{x}_j\right), \forall \mathbf{x}_i, \mathbf{x}_j \in N \\ \mathrm{i1}D_N, f(x,y) = g(x\cos\theta + y\sin\theta), \forall (x,y) \in N, f \notin \mathrm{i0}D_N \\ \mathrm{i2}D_N, \text{else} \end{cases} \tag{13}$$

where g is a 1D real-valued function.

Consider an i1D signal $f_{\mathrm{i1}D}(x, y) = g(x\cos\theta + y\sin\theta)$, where θ is its main orientation, as shown in Fig. 6a. For any point (x, y) on $f_{\mathrm{i1}D}(x, y)$, denote by g' the 1D slice obtained by cutting the $f_{\mathrm{i1}D}$ at (x, y) along the orientation θ. Then, the phase ϕ of $f_{\mathrm{i1}D}(x, y)$ is defined as the phase of the 1D signal g' evaluated at (x, y) using the analytic signal technique. With the aid of the monogenic signal, as illustrated in Fig. 6b using a spherical coordinate system, θ and ϕ can be estimated in an isotropic way as [4]

$$\theta = \arctan\frac{h_y\{f_{i1D}\}}{h_x\{f_{i1D}\}}, \theta \in [0, \pi) \tag{14}$$

$$\phi = \arctan2\left(\mathrm{sgn}(h_1\{f_{i1D}\})\sqrt{(h_x\{f_{i1D}\})^2 + \left(h_y\{f_{i1D}\}\right)^2}, f_{i1D} \right), \phi \in [0, 2\pi) \tag{15}$$

where $\mathrm{sgn}(x)$ returns the sign of x.

3.3. Higher order Riesz transforms

As stated, in case of 2D image signals, the 1st-order Riesz transform based monogenic signal enables the rotationally invariant analysis of i1D structures, such as edges and lines. In order to characterize i2D image structures, e.g., corners and junctions, higher order Riesz transforms need to be exploited [36,37]. In this paper, we only consider 2nd-order Riesz transforms. Given an image $f(\mathbf{x})$, three 2nd-order Riesz transforms of f are defined as

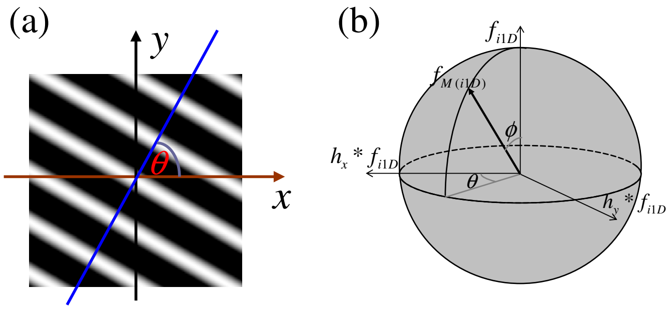

Fig. 6. (a) An example of the i1D signal $f_{\mathrm{i1}D}(x, y) = g(x\cos\theta + y\sin\theta)$, where $g = \cos(0.2\ t)$ and $\theta = \pi/3$; and (b) a geometric illustration of the monogenic signal in a spherical coordinate system.

$$h_{xx}\{f\}(\mathbf{x}) \equiv h_x\{h_x\{f\}\}(\mathbf{x}) \tag{16}$$

$$h_{xy}\{f\}(\mathbf{x}) \equiv h_x\left\{h_y\{f\}\right\}(\mathbf{x}) \tag{17}$$

$$h_{yy}\{f\}(\mathbf{x}) \equiv h_y\left\{h_y\{f\}\right\}(\mathbf{x}). \tag{18}$$

Using the convolution theorem, the transfer functions of h_{xx}, h_{xy}, and h_{yy} in the Fourier domain are

$$H_{uu}(\mathbf{u}) \equiv \left(-j\frac{u}{\sqrt{u^2+v^2}}\right)\left(-j\frac{u}{\sqrt{u^2+v^2}}\right) = -\frac{u^2}{u^2+v^2} \tag{19}$$

$$H_{uv}(\mathbf{u}) \equiv \left(-j\frac{u}{\sqrt{u^2+v^2}}\right)\left(-j\frac{v}{\sqrt{u^2+v^2}}\right) = -\frac{uv}{u^2+v^2} \tag{20}$$

$$H_{vv}(\mathbf{u}) \equiv \left(-j\frac{v}{\sqrt{u^2+v^2}}\right)\left(-j\frac{v}{\sqrt{u^2+v^2}}\right) = -\frac{v^2}{u^2+v^2}. \tag{21}$$

4. Encoding local patterns using Riesz transforms

In this section, we will present two novel coding methods, namely RCode1 and RCode2, based on the image's responses to the 1st-order and 2nd-order Riesz transforms, respectively.

4.1. Computing single-band Riesz transforms of a given image

Riesz transform based image analysis, e.g., the monogenic signal, assumes that the signal consists of few frequencies, or in other words, it is band limited. However, real images usually consist of a wide range of frequencies. Therefore, it is necessary to pre-filter the image with a chosen band-pass filter before applying the Riesz transform to it. With respect to the band-pass filter, there are many candidates proposed in the literature and we choose the widely used Gabor filter h_{BP} [6,10]. Specifically, h_{BP} adopted in this paper is of a circular isotropic shape in the spatial domain, defined as

$$h_{BP}(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x}\|_2^2}{2\varsigma^2}\right)\cos(2\pi\mu_0\|\mathbf{x}\|_2) \tag{22}$$

where μ_0 and ς are two parameters to control the shape of the Gabor filter. In real implementation, h_{BP} needs to be made zero-mean.

In order to get the Riesz transforms of the image filtered by h_{BP}, we can apply the filters h_x, h_y, h_{xx}, h_{xy} and h_{xy} to $h_{BP}\{f\}$. However, based on the associative property of the convolution operation, we can pre-compute "band-pass filtered" Riesz transform filters offline. They are defined as

$$h_{x\_BP}(\mathbf{x}) \equiv h_x\{h_{BP}\}(\mathbf{x}) \tag{23}$$

$$h_{y\_BP}(\mathbf{x}) \equiv h_y\{h_{BP}\}(\mathbf{x}) \tag{24}$$

$$h_{xx\_BP}(\mathbf{x}) \equiv h_{xx}\{h_{BP}\}(\mathbf{x}) \tag{25}$$

$$h_{xy\_BP}(\mathbf{x}) \equiv h_{xy}\{h_{BP}\}(\mathbf{x}) \tag{26}$$

$$h_{yy\_BP}(\mathbf{x}) \equiv h_{yy}\{h_{BP}\}(\mathbf{x}). \tag{27}$$

Then, we filter the image with these new filters. Obviously, with this technique, we can save one convolution operation for the online processing. Shapes of these filters (with a specific μ_0 and ς) in the spatial domain are shown in Fig. 7. Moreover, according to the formula Eq. (12), the expression for the monogenic signal of a band-pass filtered image turns to

$$f_{Mb}(\mathbf{x}) = \left(h_{BP}\{f\}(\mathbf{x}), h_{x\_BP}\{f\}(\mathbf{x}), h_{y\_BP}\{f\}(\mathbf{x})\right). \qquad (28)$$

4.2. RCode1

The first coding method proposed, RCode1, is based on the image's responses to the filters h_{BP}, h_{x_BP}, and h_{y_BP}. Specifically, given an image $f(\mathbf{x})$, its corresponding RCode1($\mathbf{x}$) is three bits obtained by binarizing $h_{BP}\{f\}(\mathbf{x})$, $h_{x\_BP}\{f\}(\mathbf{x})$, and $h_{y\_BP}\{f\}(\mathbf{x})$ according to their signs. From Eq. (28), it can be seen that $h_{BP}\{f\}(\mathbf{x})$, $h_{x\_BP}\{f\}(\mathbf{x})$, and $h_{y\_BP}\{f\}(\mathbf{x})$ are the three components of the monogenic signal $f_{Mb}(\mathbf{x})$. Thus, RCode1($\mathbf{x}$) can represent the octant of $f_{Mb}(\mathbf{x})$ in the spherical coordinate system as illustrated in Fig. 6b. Hence, RCode1($\mathbf{x}$) can roughly reflect the local orientation and the local phase of $f(\mathbf{x})$ if $f(\mathbf{x})$ is regarded locally as an i1D signal. The computation process of RCode1 is illustrated in Fig. 8 using an FKP image taken from [26].

Consider two RCode1 maps, P and Q. $P_1(Q_1)$, $P_2(Q_2)$, and $P_3(Q_3)$ are three bit-planes of $P(Q)$. In order to measure their dissimilarity, we resort to the normalized Hamming distance defined as

$$d(P,Q) = \frac{\sum_{y=1}^{Rows}\sum_{x=1}^{Cols}\sum_{i=1}^{3}P_i(x,y)\otimes Q_i(x,y)}{3S} \qquad (29)$$

where S is the area of the code map, and ⊗ represents the bitwise “exclusive OR” operation.

4.3. RCode2

In a similar fashion to deriving RCode1, by binarizing $h_{xx\_BP}\{f\}$, $h_{xy\_BP}\{f\}$ and $h_{yy\_BP}\{f\}$ according to their signs, we can get another 3-bit coding scheme RCode2 for a given image f. The computation process of RCode2 is illustrated in Fig. 9 using a palmprint image taken from [27]. Given two RCode2 maps, their dissimilarity can also be computed according to the formula in Eq. (29).

In fact, since each bit of RCode1 (or RCode2) is obtained by binarizing the image's response to a filter, it can be regarded as an ordinal measure [30] of the local image structure. Thus, RCode1 and RCode2 should have the common advantages of ordinal measures, e.g., robust to illumination changes [30].

5. Experimental results and discussions

5.1. Databases and the test protocol

Experiments were conducted on two benchmark biometric images databases, one for palmprint [27] and one for FKP [26]. In those two databases, sample images for each subject were collected in two sessions. In our experiments, we took images collected in the first session as the gallery set and images collected at the second session as the probe set. To obtain statistical results, each image in the probe set was matched with all the images in the gallery set. If the two images were from the same class, the matching between them was counted as a genuine matching; otherwise it was counted as an imposter matching. The equal error rate (EER), which is the point where the false accept rate (FAR) is equal to the false reject rate (FRR), is used to evaluate the verification accuracy. Besides, by adjusting the matching threshold, a receiver operating characteristic (ROC) curve, which is a plot of FRR against FAR for all possible thresholds, can be created. The ROC curve can reflect the overall verification

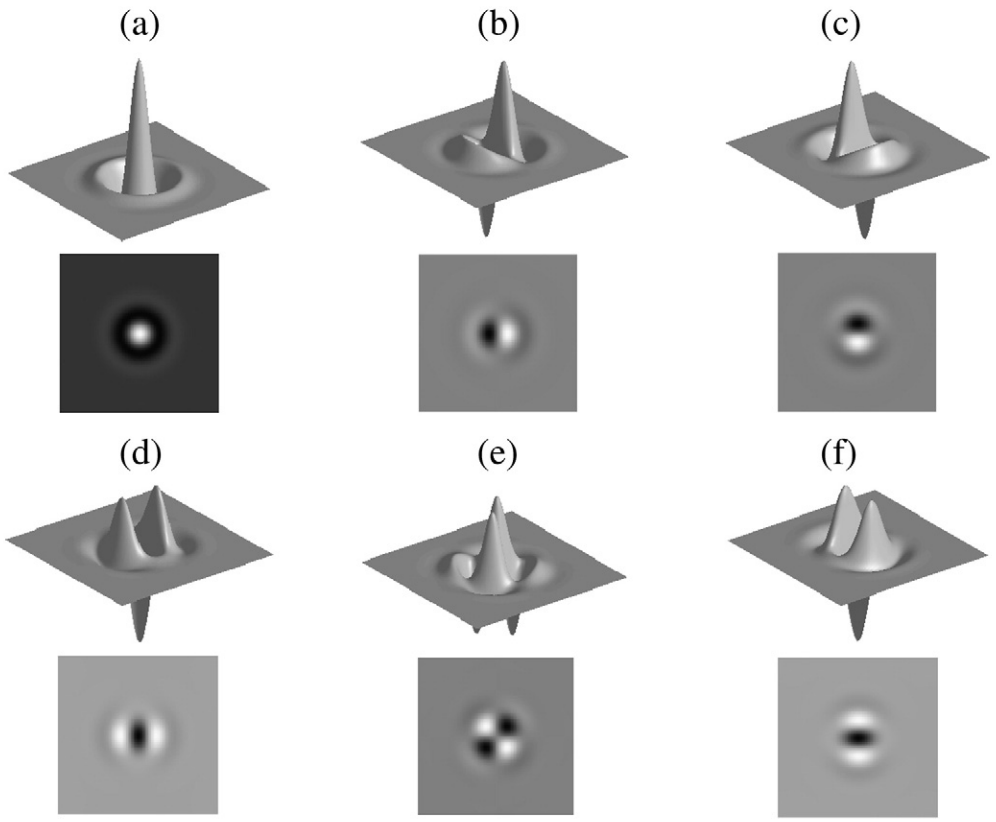

Fig. 7. Shapes of the filters in the spatial domain. They are used to calculate the Riesz transforms of an image. For each filter, its shape is shown in a 3D surface format and in a gray-scale image format. (a) h_{BP}; (b) h_{x_BP}; (c) h_{y_BP}; (d) h_{xx_BP}; (e) h_{xy_BP}; and (f) h_{yy_BP}.

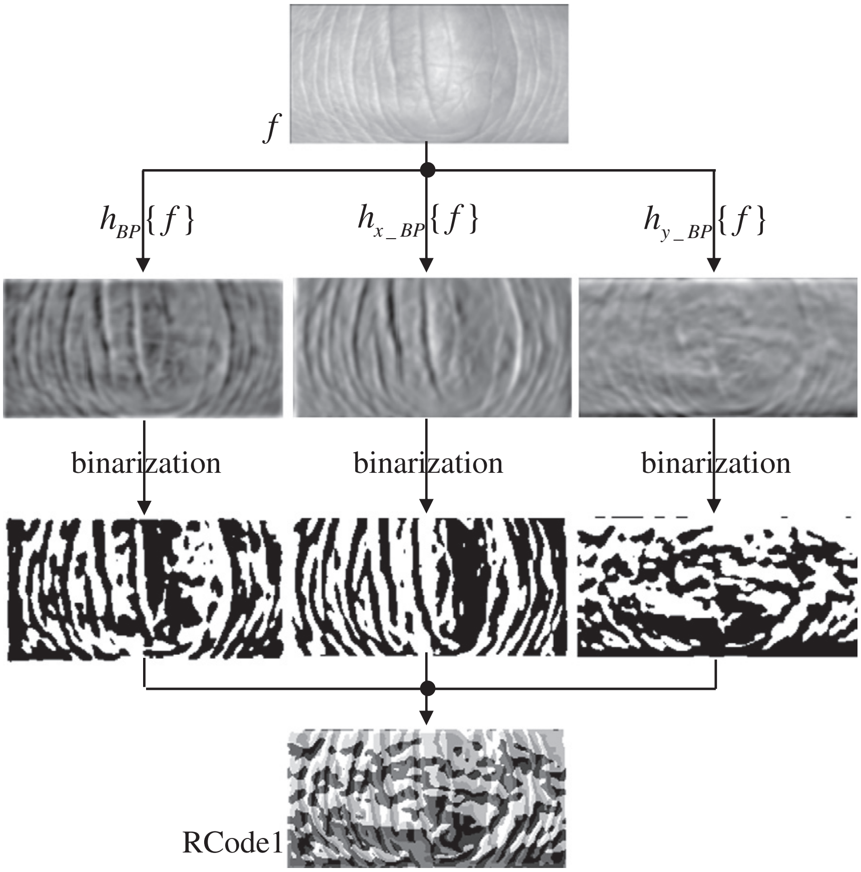

Fig. 8. Illustration for the RCode1 computation process for a given finger-knuckle-print image.

accuracy of a biometric system. Thus, the ROC curve obtained by using each evaluated method will be provided. We evaluated and compared the performance of five coding-based methods, including CompCode, RLOC, OrdinalCode, RCode1 and RCode2.

5.2. Palmprint verification

The PolyU palmprint database [27] contains 7752 grayscale palmprint images collected from 386 different palms. Each image is of the size 384×284 pixels. For each palm, there are around 20 samples collected in two sessions, where around 10 samples were collected for each session. The average time interval between the first and second sessions was about 2 months. An ROI extraction procedure similar to that in [42] was used to extract the palmprint ROI of the size 128×128 pixels. Under our experimental settings, the gallery set contained 3889 images while the probe set contained 3863 images. The numbers of genuine matchings and imposter matchings were 38,924 and 14,949,372, respectively. To reduce the negative effect of imperfect ROI extraction, we shifted the code maps vertically and horizontally in a small range when matching. The minimal distance obtained by shift matching was taken as the final distance. The shift range was set as [−3, 3] in the following experiments.

For each evaluated coding method, there are some parameters to be set. To this end, we tuned the parameters on a sub-dataset, which contained the first 193 palms. Parameter settings and verification accuracies for all the evaluated methods are listed in Table 1. Fig. 10 shows the ROC curves generated by the five coding schemes.

From Table 1 and Fig. 10, we can see that the proposed method RCode2 could achieve a similar performance with CompCode, while they both perform much better than the other methods evaluated in terms of the verification accuracy for palmprint verification.

5.3. FKP verification

In PolyU FKP database [26], images were collected from 165 volunteers, including 125 males and 40 females. Samples were collected in two separate sessions. In each session, the subject was asked to provide 6 images for each of the left index finger, the left middle finger, the right index finger and the right middle finger. Therefore, 48 images from 4 fingers were collected from each subject. In total, the database contains 7920 images from 660 different fingers. The average time interval between the first and the second sessions was about 25 days. ROI images, which are also available at [26], were extracted using the algorithm described in [46]. Under our experimental settings, the gallery set and the probe set both contained 3960 images. The numbers of genuine matchings and imposter matchings were 23,760 and 15,657,840, respectively.

A sub-dataset, which contained the first 300 FKP classes, was used to adjust the parameters required for each evaluated method. Parameter settings and verification accuracies for all the evaluated methods are summarized in Table 2. Fig. 11 shows the ROC curves generated by the five coding schemes.

From Table 2 and Fig. 11 we can see that the proposed coding methods RCode1 and RCode2 can achieve similar performance with CompCode, while they are much better than the other methods evaluated.

5.4. Discussions

In Sections 5.2 and 5.3, we compared the performance of five coding based methods for palmprint verification and FKP verification. Since all of the five methods use 3-bits to represent each code, they all have the same feature size. Besides, since they all use the normalized Hamming distance at the matching stage, they have the same

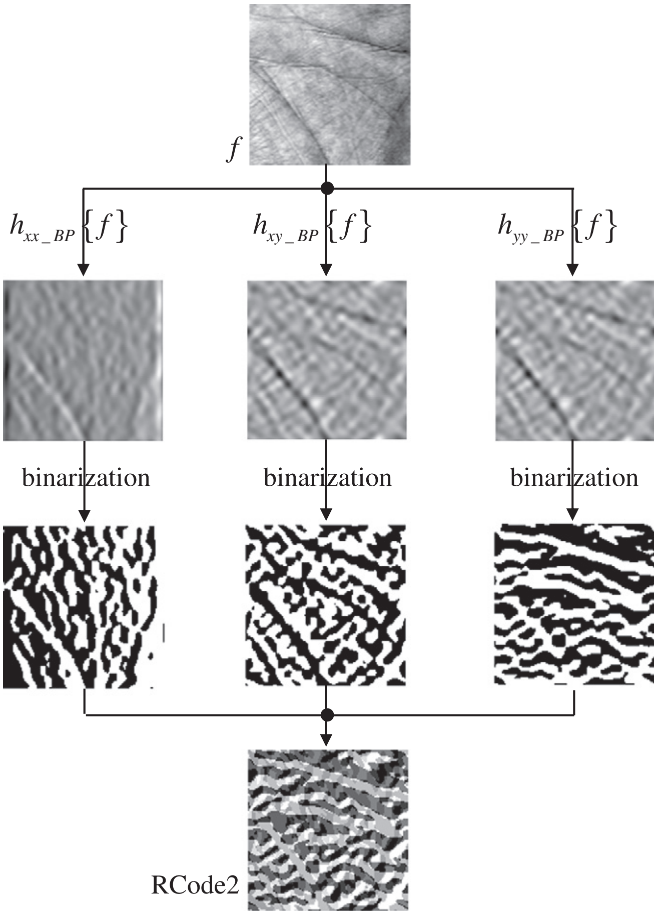

Fig. 9. Illustration for the RCode2 computation process for a given palmprint image.

computational cost for matching. With respect to the verification accuracy, CompCode and RCode2 have quite similar performances and they are more stable and better than the other methods evaluated. However, these two methods have different computational complexities at the feature extraction stage. Since the major operations at the feature extraction stage are the convolutions, the number of convolutions can roughly reflect the overall feature extraction complexity. RCode2 needs three convolutions while CompCode needs six. In fact, for CompCode, an extra operation is needed to figure out the minimum of the filters' responses at each position. Table 3 lists the number of convolutions needed by each method for feature extraction. Besides, in Table 3, for each method, we also present the time consumption for the feature extraction for one FKP image. Each method was implemented with Visual C#.Net 2005 on a Dell Inspiron 530s PC embedded Intel 6550 processor and 2 GB RAM. Convolutions were accomplished via the FFT transform according to the convolution theorem. We can see that RCode2 is nearly three times faster than CompCode at the feature extraction stage. Thus, we claim that the proposed RCode2 method is the best 3-bit coding method and can serve as a better candidate for real-time applications.

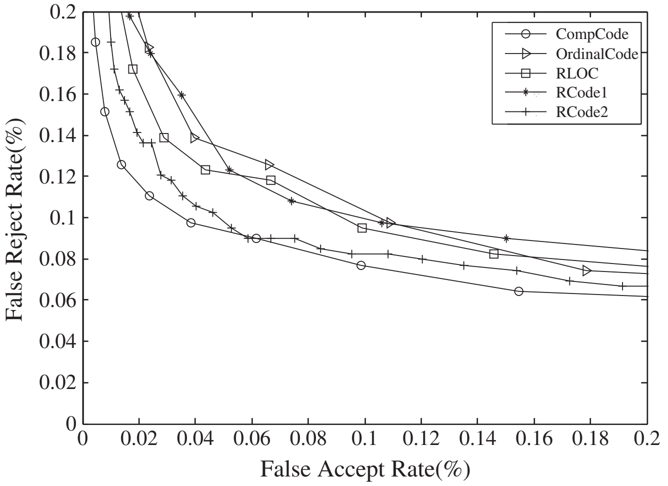

Fig. 10. ROC curves obtained by the five coding methods on PolyU palmprint database.

It needs to be pointed out that, if only the verification accuracy is considered, there are even better methods existing in the literature. However, compared with the 3-bit coding schemes evaluated in our experiments, these methods usually has a much larger feature size and need more time for feature extraction and matching. For example, to our knowledge, the method that can achieve the highest verification accuracy for palmprint verification is BOCV [7]. Under the same experimental settings as described in Section 5.2, BOCV can achieve an EER 0.071%, lower than RCode2. However, BOCV needs 6-bits to represent each code, so the size of its feature map is twice as the 3-bit coding schemes; and accordingly, it needs longer time for feature matching. For FKP verification, under the same experimental settings as described in Section 5.3, OE-SIFT [24] can get an EER 0.85%, lower than RCode2. However, OE-SIFT works much slower than the coding-based methods since it depends on images' SIFT features which are quite time-consuming to extract. So, we claim that RCode2 can actually achieve a desired balance on several performance metrics, including the verification accuracy, the feature size, the feature extraction speed, and the feature matching speed.

6. Conclusions

Recent studies have shown that Riesz transforms can capture the local image structures powerfully. In this paper, we introduced them into the biometrics community by proposing two novel coding-based feature extraction methods for palmprint and finger-knuckle-print recognition, namely RCode1 and RCode2, by binarizing the image's responses to the Riesz transforms. Both of these two methods use 3-bits to represent each code and adopt the normalized Hamming distance for matching. Performances of the proposed coding schemes were evaluated and compared with the other 3-bit coding methods on a palmprint database and a finger-knuckle-print database. Experimental results corroborated that both the methods, especially RCode2, could achieve high verification accuracies. Particularly, RCode2 could achieve quite similar verification accuracy with the state-of-the-art

Table 1
Evaluation of five coding methods for palmprint verification.

	Key parameter settings	EER (%)
CompCode [15]	$\sigma=4.7$, $\delta=1.39$	0.083
RLOC [14]	$w=4$	0.116
OrdinalCode [31]	$\sigma_x=1.5$, $\sigma_y=6.0$	0.101
RCode1	$\mu_0=0.087$, $\varsigma=4.42$	0.105
RCode2	$\mu_0=0.086$, $\varsigma=4.55$	0.085

Table 2
Evaluation of five coding methods for FKP verification.

	Key parameters settings	EER (%)
CompCode [15]	$\sigma=5.3$, $\delta=3.32$	1.658
RLOC [14]	$w=4$	1.912
OrdinalCode [31]	$\sigma_x=1.2$, $\sigma_y=4.2$	3.405
RCode1	$\mu_0=0.085$, $\varsigma=4.82$	1.661
RCode2	$\mu_0=0.084$, $\varsigma=5.30$	1.610

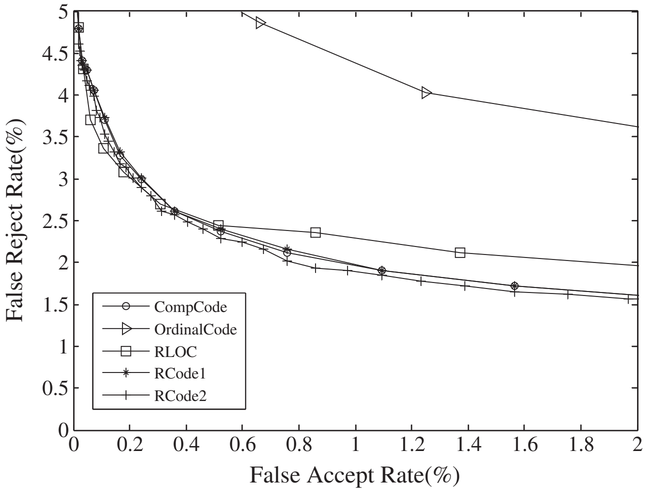

Fig. 11. ROC curves obtained by the five coding methods on PolyU finger-knuckle-print database.

Table 3
Computational cost of five coding schemes at the extraction stage.

	Number of convolutions	Extraction time (ms) for 1 FKP image
CompCode [15]	6	54.6
RLOC [14]	6	54.6
OrdinalCode [31]	3	17.6
RCode1	3	17.6
RCode2	3	17.6

coding method CompCode; compared with CompCode, however, RCode2 is nearly 3 times faster for feature extraction. So we conclude that RCode2 is the best 3-bit coding-based feature extraction method for both the palmprint recognition and the finger-knuckle-print recognition.

Acknowledgment

This work is supported by the Fundamental Research Funds for the Central Universities under grant no. 2100219033, the Natural Science Foundation of China under grant no. 60903120 and no. 61201394, the Shanghai Natural Science Foundation under grant no. 09ZR1434400, and the Innovation Program of Shanghai Municipal Education Commission under grant no. 12ZZ029.

References

[1] T. Connie, A.T.B. Jin, M.G.K. Ong, D.N.C. Ling, An automated palmprint recognition system, Image Vis. Comput. 23 (5) (2005) 501–515.
[2] J.G. Daugman, High confidence visual recognition of persons by a test of statistical independence, IEEE Trans. Pattern Anal. Mach. Intell. 15 (11) (1993) 1148–1161.
[3] N. Duta, A survey of biometric technology based on hand shape, Pattern Recognit. 42 (11) (2009) 2797–2806.
[4] M. Felsberg, G. Sommer, The monogenic signal, IEEE Trans. Signal Process. 49 (12) (2001) 3136–3144.
[5] M. Felsberg, G. Sommer, The monogenic scale-space: a unifying approach to phase-based image processing in scale space, J. Math. Imaging Vis. 21 (1) (2004) 5–26.
[6] D. Gabor, Theory of communication, J. Inst. Electr. Eng. 93 (III) (1946) 429–457.
[7] Z. Guo, D. Zhang, L. Zhang, W. Zuo, Palmprint verification using binary orientation co-occurrence vector, Pattern Recognit. Lett. 30 (13) (2009) 1219–1227.
[8] S.L. Hahn, Hilbert Transforms in Signal Processing, Artech House, Boston, 1996.
[9] S. Held, M. Storath, P. Massopust, B. Forster, Steerable wavelet frames based on the Riesz transform, IEEE Trans. Image Process. 19 (3) (2010) 653–667.
[10] L. Hong, Y. Wan, A. Jain, Fingerprint image enhancement: algorithm and performance evaluation, IEEE Trans. Pattern Anal. Mach. Intell. 20 (8) (1998) 777–789.
[11] A.K. Jain, J. Feng, Latent palmprint matching, IEEE Trans. Pattern Anal. Mach. Intell. 31 (6) (2009) 1032–1047.
[12] A.K. Jain, P.J. Flynn, A. Ross, Handbook of Biometrics, Springer, 2007.
[13] W. Jia, D. Huang, D. Zhang, Palmprint verification based on principal lines, Pattern Recognit. 41 (4) (2008) 1316–1328.
[14] W. Jia, D. Huang, D. Zhang, Palmprint verification based on robust line orientation code, Pattern Recognit. 41 (5) (2008) 1504–1513.
[15] A. Kong, D. Zhang, Competitive coding scheme for palmprint verification, in: Proceedings of the ICPR'04, 2004, pp. 520–523.
[16] A. Kong, D. Zhang, M. Kamel, A survey of palmprint recognition, Pattern Recognit. 42 (7) (2009) 1408–1418.
[17] A. Kumar, C. Ravikanth, Personal authentication using finger knuckle surface, IEEE Trans. Inf. Forensics Secur. 4 (1) (2009) 98–109.
[18] A. Kumar, Y. Zhou, Personal identification using finger-knuckle orientation features, Electron. Lett. 45 (20) (2009) 1023–1025.
[19] T.S. Lee, Image representation using 2D Gabor wavelet, IEEE Trans. Pattern Anal. Mach. Intell. 18 (10) (1996) 957–971.
[20] M.K.H. Leung, A.C.M. Fong, S.C. Hui, Palmprint verification for controlling access to shared computing resources, IEEE Pervasive Comput. 6 (4) (2007) 40–47.
[21] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[22] D. Maltoni, D. Maio, A.K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, Springer, 2003.
[23] M. Mellor, M. Brady, Phase mutual information as a similarity measure for registration, Med. Image Anal. 9 (4) (2005) 330–343.
[24] A. Morales, C.M. Travieso, M.A. Ferrer, J.B. Alonso, Improved finger-knuckle-print authentication based on orientation enhancement, Electron. Lett. 47 (6) (2011) 380–381.
[25] L. Moura, P. Monteiro, Design method for FIR-based Hilbert transform filters suitable for broadband AM-SSB, Electron. Lett. 38 (12) (2002) 605–606.
[26] PolyU Finger-Knuckle-Print Database, http://www.comp.polyu.edu.hk/~biometrics 2010.
[27] PolyU Palmprint Database, http://www.comp.polyu.edu.hk/~biometrics 2006.
[28] L. Shang, D. Huang, J. Du, C. Zheng, Palmprint recognition using FastICA algorithm and radial basis probabilistic neural network, Neurocomputing 69 (13) (2006) 1782–1786.
[29] E. Stein, G. Weiss, Introduction to Fourier Analysis on Euclidean Spaces, Princeton University Press, Princeton, NJ, 1971.
[30] Z. Sun, T. Tan, Ordinal measures for iris recognition, IEEE Trans. Pattern Anal. Mach. Intell. 31 (12) (2009) 2211–2226.
[31] Z. Sun, T. Tan, Y. Wang, S.Z. Li, Ordinal palmprint representation for personal identification, in: Proceedings of the CVPR'05, 2005, pp. 279–284.
[32] K. Takaya, Feature point correspondence of stereo images by monogenic phase, in: Proceedings of the PacRim'07, 2007, pp. 272–275.
[33] M. Unser, D. Sage, D.V.D. Ville, Multiresolution monogenic signal analysis using the Riesz–Laplace wavelet transform, IEEE Trans. Image Process. 18 (11) (2009) 2402–2418.
[34] M. Unser, D.V.D. Ville, Wavelet steerability and the higher-order Riesz transform, IEEE Trans. Image Process. 19 (3) (2010) 636–652.
[35] J. Wang, W. Yau, A. Suwandy, E. Sung, Personal recognition by fusing palmprint and palm vein images based on "Laplacianpalm" representation, Pattern Recognit. 41 (5) (2008) 1531–1544.
[36] L. Wietzke, G. Sommer, The 2D analytic signal, in: Technical Report, Kiel University, 2008.
[37] L. Wietzke, G. Sommer, The signal multi-vector, J. Math. Imaging Vis. 37 (2) (2010) 132–150.
[38] D.L. Woodard, P.J. Flynn, Finger surface as a biometric identifier, Comput. Vision Image Understanding 100 (3) (2005) 357–384.
[39] X. Wu, D. Zhang, K. Wang, Fisherpalms based palmprint recognition, Pattern Recognit. Lett. 24 (15) (2003) 2829–2838.
[40] W. Yang, C. Sun, L. Zhang, A multi-manifold discriminant analysis method for image feature extraction, Pattern Recognit. 44 (8) (2011) 1649–1657.
[41] D. Zang, G. Sommer, Signal modeling for two-dimensional image structures, J. Visual Commun. Image Represent. 18 (1) (2007) 81–99.
[42] D. Zhang, W. Kong, J. You, M. Wong, Online palmprint identification, IEEE Trans. Pattern Anal. Mach. Intell. 25 (9) (2003) 1041–1050.
[43] L. Zhang, L. Zhang, Z. Guo, D. Zhang, Monogenic-LBP: a new approach for rotation invariant texture classification, in: Proceedings of the ICIP'10, 2010, pp. 2677–2680.
[44] L. Zhang, L. Zhang, D. Zhang, Finger-knuckle-print: a new biometric identifier, in: Proceedings of the ICIP'09, 2009, pp. 1981–1984.
[45] L. Zhang, L. Zhang, D. Zhang, H. Zhu, Ensemble of local and global information for finger-knuckle-print recognition, Pattern Recognit. 44 (9) (2011) 1990–1998.
[46] L. Zhang, L. Zhang, D. Zhang, H. Zhu, Online finger-knuckle-print verification for personal authentication, Pattern Recognit. 43 (7) (2010) 2560–2571.