Machine Intelligence Research
www.mi-research.net

21(3), June 2024, 597-614
DOI: 10.1007/s11633-023-1436-6

Enhance the Performance of Directional Feature-based Palmprint Recognition by Directional Response Stability Measurement

Haitao Wang Wei Jia

School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China

Abstract: Palmprint recognition is an emerging biometrics technology that has attracted increasing attention in recent years. Many palmprint recognition methods have been proposed, including traditional methods and deep learning-based methods. Among the traditional methods, the methods based on directional features are mainstream because they have high recognition rates and are robust to illumination changes and small noises. However, to date, in these methods, the stability of the palmprint directional response has not been deeply studied. In this paper, we analyse the problem of directional response instability in palmprint recognition methods based on directional feature. We then propose a novel palmprint directional response stability measurement (DRSM) to judge the stability of the directional feature of each pixel. After filtering the palmprint image with the filter bank, we design DRSM according to the relationship between the maximum response value and other response values for each pixel. Using DRSM, we can judge those pixels with unstable directional response and use a specially designed encoding mode related to a specific method. We insert the DRSM mechanism into seven classical methods based on directional feature, and conduct many experiments on six public palmprint databases. The experimental results show that the DRSM mechanism can effectively improve the performance of these methods. In the field of palmprint recognition, this work is the first in-depth study on the stability of the palmprint directional response, so this paper has strong reference value for research on palmprint recognition methods based on directional features.

Keywords: Biometrics, palmprint recognition, directional response stability, directional coding-based methods, directional feature.

Citation: H. Wang, W. Jia. Enhance the performance of directional feature-based palmprint recognition by directional response stability measurement. *Machine Intelligence Research*, vol.21, no.3, pp.597–614, 2024. http://doi.org/10.1007/s11633-023-1436-6

1 Introduction

Palmprint recognition, as a promising biometric recognition technology, has attracted wide attention from academia and industry. To date, researchers have proposed a number of palmprint recognition methods[1], which can be divided into two categories, i.e., traditional methods[2–4] and deep learning-based methods[5, 6]. In recent years, deep learning has made breakthrough progress in many tasks of computer vision[7–9]. However, deep learning technology requires a large amount of training data and computing resources. In palmprint recognition, traditional methods can achieve stable and good recognition results on some databases with limited data and computing resources. Therefore, traditional methods still have high research and application value. The traditional methods can be further divided into the following subcategories: palm line-based methods, directional coding-based methods, texture-based methods[10, 11], subspace learning methods and correlation filter-based methods. Palm lines, including principal lines and wrinkles, are the basic elements of palmprints.

Palm lines contain obvious directional features, which are robust to illumination changes and noise, so directional feature-based methods have achieved good recognition performance. According to the representation and matching modes of directional features, directional feature-based methods can be further divided into directional coding-based methods, directional histogram-based methods, and combining direction and correlation filter-based methods.

Directional coding-based methods first extract the direction map of a palmprint image. Then, the direction map is downsampled, and the direction value of each pixel after downsampling is binary coded. Finally, the coded directional features of different palmprints are matched by Hamming distance. This kind of method can effectively reduce the computational complexity and storage cost. Classical directional coding methods mainly include competitive code (CompC)[12], ordinal code, binary

Research Article
Manuscript received on February 6, 2023; accepted on March 13, 2023; published online on February 22, 2024
Recommended by Associate Editor Hao Dong
Colored figures are available in the online version at https://link.springer.com/journal/11633

orientation co-occurrence vector (BOCV)[13], double-orientation code (DOC),[14] and discriminative and robust competitive code (DRCC)[15], etc.

Directional histogram methods first extract the directional map of the palmprint image, then divide the directional map into small blocks, calculate the histograms based on directional features on the small blocks, and finally connect the histograms of all small blocks to form the final histogram. Classic directional histogram methods include histogram of oriented lines (HOL)[16], local line directional pattern (LLDP)[17], and collaborative representation-based competitive code (CR_CompC)[18], etc.

Combining direction and correlation filter-based methods are performed in the frequency domain. This kind of method first obtains the direction map of the palmprint image, then extracts the frequency domain features from the direction map and matches them in the frequency domain. Typical methods include complete direction representation (CDR) method[19] based on band-limited phase-only correlation (BLPOC).

All methods based on directional features first need to calculate the direction map of the palmprint image. To calculate the direction map, a group of line-like filters, such as Gabor filters, are generally used to filter the palmprint region of interest (ROI) image, and then the direction of each pixel is determined according to the magnitude relationship of the filter response values of this group of filters. Furthermore, researchers have proposed a series of strategies to improve the robustness and recognition accuracy of the method based on directional features. The main strategies include trying to use different filters, increasing the number of filters, exploiting multiscale information, using feature learning to obtain more discriminative directions, using multiple directions to modulate features, considering the magnitude of domain response, or using different types of classifiers, etc.

Through the analysis of directional feature-based palmprint recognition methods, we found that the stability of directional features of palmprints has not been deeply studied. In the research of CompC[7], we found that when the difference between a set of response values is small or the advantage of the maximum response value relative to other response values is insufficient, the available direction information easily fluctuates, which will lead to a change in direction feature selection.

Fig. 1 shows two ROI images collected in two different sessions from the same person. According to the operation in CompC[7], Gabor filters in six directions are convolved at each position, and responses in six directions $(\text{dir}1, \cdots, \text{dir}6)$ are obtained. CompC uses the winner-take-all rule to determine the directional feature of each pixel. For Fig. 1(a), the direction of the orange position is determined as 5 (dir5), and the direction of the blue position is determined as 3 (dir3). For Fig. 1(b), the direction of the orange position is determined as 5 (dir5), and the direction of the blue position is determined as 4 (dir4).

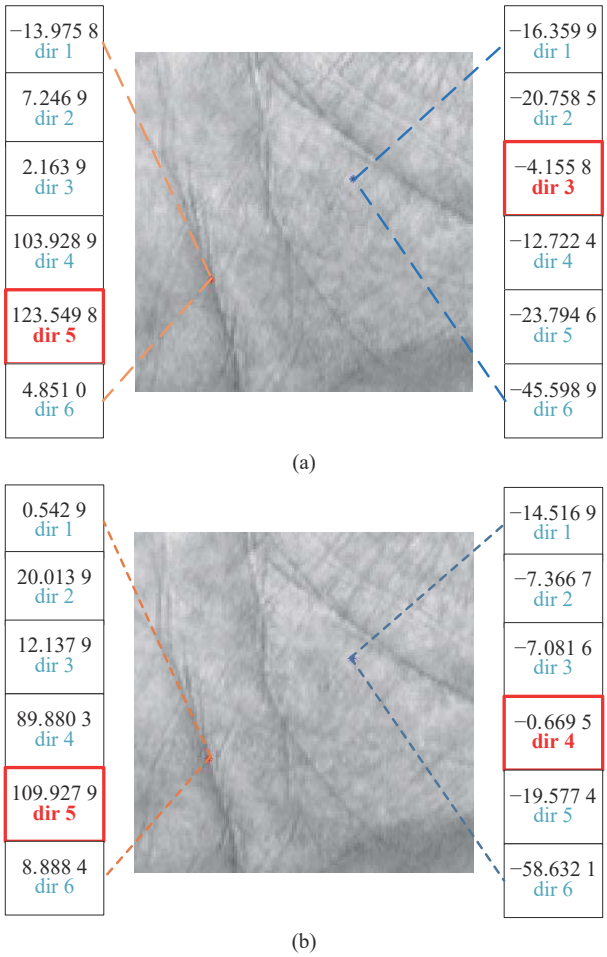

Fig. 1 Examples of stability of palmprint directional features, where two ROI images were collected in two different sessions from the same person. (a) Filter response values of two pixels with different positions in the first image; (b) Filter response values of two pixels with different positions in the second image.

Because two ROI images are collected from the same person, we hope that the directional features of two samples in blue position should be the same. Obviously, in two samples, there is a difference in blue positions between two samples, which will affect the recognition result.

The stability of a set of directional responses of a pixel may be affected by many factors. One of the main factors is that the pixel is located in an area with little gray change because the direction filter is usually used to detect the gray change of a region, so the direction feature is not obvious. Other factors include lighting changes, noise, blurred images and hand posture changes.

In the research of iris recognition, in the modelling and analysing of the false acceptance rate (FAR) and false reject rate (FRR) of iris recognition systems, Bolle et al.[20] found that the theoretical accuracy of FRR deteriorates rapidly when the bit inversion rate increases, and the degradation is more obvious than the theoretical prediction, which means that the invariant bits in iris code are significantly robust to image noise. Bolle et al.[20]

guessed that the reason for this phenomenon is that not all bits are equally stable. The so-called stability means having the same probability of flipping, and some possible bits may be particularly easy to flip. Bolle et al.[20] called these bits fragile bits. Later, Hollingsworth et al.[21] performed a scientific analysis of the fragile bit phenomenon, confirmed Bolle′s conjecture through experiments, and specifically pointed out that the fragile bit phenomenon easily occurs in different genders, filters and different regions in the iris, indicating that the fragile bit will affect the selection invariance of features, and this influence cannot be eliminated by matching translation or rotation features. Therefore, Hollingsworth et al.[21] proposed the fragile bits masking strategy, which can reduce the side effects of fragile bits in the feature matching process by covering up some features that are prone to fragile bit positions.

In palmprint recognition, Zhang et al.[22] proposed the method of BOCV (E-BOCV), which directly applied the fragile bits masking strategy to BOCV, a classical palmprint recognition method based on directional features. Except for the E-BOCV, other researchers have never discussed the stability of palmprint directional responses.

The direction of the maximum response is the most valuable information in a group of responses[12, 14, 15] because it largely reflects the directional feature of the palmprint in the area where the sampling pixels are located. Generally, for a pixel, when the difference between a group of directional responses is small or the difference between the value of the largest response and the value of the second largest response is small, we think that the directional feature extracted from this group of responses is unstable.

In this paper, we propose a novel palmprint directional response stability measurement (DRSM) of pixels. Based on DRSM, we propose a new recognition framework for methods based on directional features. This framework contains the judgment mechanism of the stability of the directional response of each pixel. For those pixels with unstable directional responses, we carry out special processing to improve the recognition performance of methods based on directional features. For those pixels with unstable directional values, the general idea is to discard them. But we think that these pixels can also become the unique attributes of each palmprint image and can be converted into the same exploitable and distinctive features as normal points. In this way, the potential of coding methods based on directional features can be fully tapped. Fig. 2 shows a flow chart of a general coding method based on directional features combined with DRSM. Given a palmprint ROI image, we use Gabor filters in K directions to filter it. Then we can obtain responses in K directions. For a pixel, a DRSM module is added to determine whether the directional response of the pixel is stable. If the DRSM module judges that the directional response of the pixel is stable, the directional response of this pixel will be coded in the normal way; otherwise, it will be coded in a special way.

The main contributions of this paper are as follows:

1) We summarize the development of palmprint recognition methods based on directional features so that readers can have a deeper and more comprehensive understanding of palmprint recognition methods based on directional features.

2) We analyse the directional response instability in the palmprint recognition method based on directional features in detail and propose a palmprint DRSM. Based on DRSM, we further propose a new recognition framework for directional feature-based palmprint recognition methods. The stability of the directional response of palmprints is seldom studied. The work of this paper is the first time in the field of palmprint recognition to deeply analyse the stability of the palmprint directional response and propose an effective solution. Therefore, this work has made a significant and special contribution to palmprint recognition based on directional features.

3) DRSM is very flexible and free in use and design. The fragile bits masking strategy[21] proposed in iris recognition can only be applied to specific feature coding methods. DRSM is not limited by specific feature coding forms and can be easily applied to various methods. We insert DRSM into seven classical methods based on directional features and conduct many experiments on six public palmprint databases. Experimental results show that DRSM can effectively improve the performance of these methods.

The rest of this paper is organized as follows. Section 2 presents the development of palmprint recognition methods based on directional features. Section 3 introduces the DRSM methodology. Section 4 introduces how to insert DRSM into seven classical palmprint recognition methods based on directional features. Section 4 reports the experimental results. In Section 5, we discuss some design principles and design choices of DRSM in detail. In Section 6, the conclusions are given.

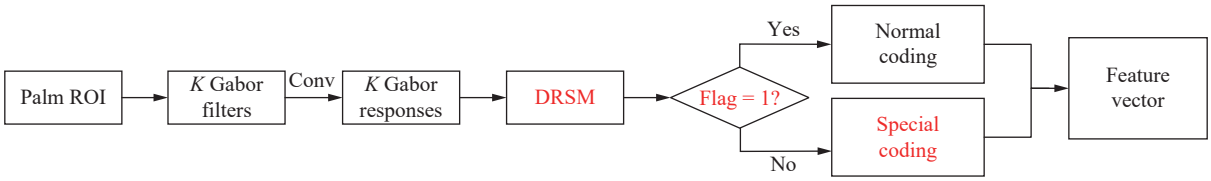

Fig. 2 The flow chart of a general coding method based on directional feature combined with DRSM

2 Related work

2.1 Directional feature-based palmprint recognition methods

To better understand the development of recognition methods based on directional features in the field of palmprint recognition, some details of recognition methods based on directional features are listed in Table 1 according to the method type and the proposed time. As we mentioned above, directional feature-based methods can be divided into directional coding-based methods and directional histogram-based methods and combined direction and correlation filter-based methods. Thus, in Table 1, the method types of directional feature-based methods are marked as "coding", "histogram", and "correlation filter".

In 2003, Kong et al.[23] and Zhang et al.[24] proposed the PalmCode method. PalmCode can be regarded as the first method to encode palmprint directional features, in which a circular Gabor filter with a certain direction is used to filter the palmprint ROI image, and then the signs of the filtered images are coded as a feature vector. In the matching stage, two PalmCodes are matched by using the normalized Hamming distance. In 2004, Kong and Zhang[12] proposed the method of CompC, in which the real part of the Gabor filters in six directions are used to filter the images of palmprint ROI. In each pixel, the indexing number of the minimum response obtained by the winner-take-all rule is treated as the directional feature. To match more efficiently, CompC encodes the indexing number representing the directional feature into 4 bit-planes. Kong and Zhang[12] also proposed the angular distance to measure two CompCs. In 2005, Wu et al.[25] proposed a method similar to CompC, named palmprint orientation code (POC), in which directional templates with 4 directions are used to define the direction of each pixel, and Hamming distance is used to measure the similarity of two POCs.

In 2005, Sun et al.[26] proposed the ordinal code method, in which three pairs of orthogonal Gaussian filters are used to filter the palmprint ROI image. Based on the ordinal measure, each pixel gets 3 bit codes. The matching of ordinal code is also based on the Hamming distance. In 2006, Kong et al.[27] proposed the method of fusion code, an improved version of palm code. In fusion code, elliptical Gabor filters with 4 directions are first used to filter the palmprint ROI image, and then, the magnitude of filtering responses is used for information fusion, and the phase is used for encoding the final feature. In 2008, Zuo et al.[28] extended competitive code to multiscale competitive code (MCC), in which 2D Log-Gabor filters with 2 scales and 6 directions are used for extracting direction coding, and a hierarchical matching scheme was proposed. In the same year, Jia et al.[29] proposed the method of robust line orientation code (RLOC). RLOC uses a modified finite Radon transform (MFRAT) to obtain the dominant direction of each pixel and uses a pixel-to-area comparison algorithm for matching. Because MFRAT accumulates pixel values along lines in different directions, it has a very fast speed when calculating directional features. In the original CompC, the directions of filters are uniformly distributed. Yue et al.[30] proposed an improved CompC, in which optimal filtering directions are calculated by a fuzzy C-means clustering algorithm. In 2019, Guo et al.[13] noticed that using only one dominant direction to represent a local region may lose some valuable information because there are cross lines in the palmprint. Therefore, they proposed the method of binary orientation co-occurrence vector (BOCV), which calculates a 6-bit binarized vector by concatenating the normalized responses along 6 directions[13]. Zuo et al.[31] extended the MCC method to sparse multiscale competitive code (SMCC). The SMCC uses the 11-norm sparse coding to obtain a robust estimation of the multiscale orientation field by using a filter bank of second derivatives of Gaussians with 6 different directions and 3 scales.

In 2011, Khan et al.[32] proposed the method of contour code, in which the non-subsampled contourlet transform (NSCT) was used to obtain the dominant directions of each pixels. Zhang et al.[22] analysed the fragile bits phenomenon and then extended BOCV to E-BOCV by incorporating fragile bits information. Sun et al.[33] proposed the method of linear programming ordinal measures (LP-OM) using a feature selection mechanism, which is an improved version of the ordinal code. Fei et al.[14] also noticed that it may not be very robust to use only one indexing number of the dominant response as the directional feature. Therefore, they proposed to use the two maximum response directions for coding named the double-orientation code (DOC)[14]. Because some palm lines are curves, using the Gabor filter bank may not accurately detect the direction of a pixel located on these curves. Fei et al.[34] proposed the method of half orientation code, which uses two half Gabor filters to fit the curves better. In 2016, Fei et al.[35] proposed the neighboring direction indicator (NDI) method, which can not only represent the most dominant directional feature of the palmprint but also denote the multiple directions of some special points having double dominant directions. To better detect the dominant direction of pixels located in the curves, Tabejamaat and Mousavi[36] proposed using a banana filter instead of a Gabor filter.

In 2016, Zheng et al.[37] proposed the fast-competitive code (Fast-CompC) method, which uses one pair of orthogonal Gabor filters to filter the palmprint ROI image and then obtains 1 bit code by the ordinal measure. Because fast-competitive code only uses two Gabor filters to filter palmprint ROI images and only uses 1 bit to represent the directional feature, it has a fast processing speed. In 2018, Xu et al.[15] proposed the method of discriminat-

Table 1 List of directional feature-based palmprint recognition methods

Method type	Method name	Filter type	Filter number	Scales	Direction levels	Mechanism for obtaining direction representation	Matching strategy	Ref	Year
Coding	PalmCode	Circular Gabor filter	1	1	1	One direction filtering	Hamming distance	[23, 24]	2003
	Competitive code	Gabor filter	6	1	1	Winner-take-all	Angular distance	[12]	2004
	POC	Directional filter	4	1	1	Winner-take-all	Hamming distance	[25]	2005
	Ordinal code	Gaussian filter	3 pairs	1	1	Ordinal measure	Hamming distance	[26]	2005
	Fusion code	Elliptical Gabor filter	4	1	1	Winner-take-all and phase coding	Hamming distance	[27]	2006
	MCC	2D Log-Gabor filter	12	2	1	Winner-take-all	Angular distance	[28]	2008
	RLOC	MFRAT	6	1	1	Winner-take-all	Pixel-to-area matching	[29]	2008
	Improved competitive code	Circular Gabor filter	6	1	1	Winner-take-all	Angular distance	[30]	2009
	BOCV	Gabor filter	6	1	1	Six direction filtering	Hamming distance	[13]	2009
	SMCC	Second derivatives of Gaussian filter	18	3	1	Winner-take-all	Angular distance	[31]	2010
	Contour code	Contourlet	8	1	1	Winner-take-all	Binary hash table	[32]	2011
	E-BOCV	Gabor filter	6	1	1	Six direction filtering	Hamming distance	[22]	2011
	LP-OM	Multi-lobe ordinal filters	Random setting	Random setting	1	Ordinal measure	Hamming distance	[33]	2013
	DOC	Gabor filter	6	1	2	Responses′ sorting	Nonlinear angular matching	[14]	2016
	HOC	Half-Gabor filters	6	1	1	Winner-take-all	Hamming distance	[34]	2016
	NDI	Circular Gabor filter	6	1	1	Winner-take-all	Hamming distance	[35]	2016
	Concavity-orientation coding	Banana filter	3 pairs	1	1	Winner-take-all	Hamming distance	[36]	2016
	Fast-competitive code	Gabor filter	1 pair	1	1	Ordinal measure	Hamming distance	[37]	2016
	DRCC	Circular Gabor filter	6	1	1	Winner-take-all	Hamming distance	[15]	2018
	MOSDL	Gabor filter bank	24	4	1	Feature learning	Hamming distance	[38]	2019
	DVDM	Circular Gabor filter	6	1	1	Winner-take-all	Hamming distance	[39]	2020
	EDM	Circular Gabor filter	6	1	1	Winner-take-all	Hamming distance	[40]	2020
Histogram	HOL	Gabor filter	12	1	1	Winner-take-all	Euclidean distance	[16]	2014
	LLDP	Gabor filter	12	1	2	Responses′ sorting	Chi-square distance	[17]	2016
	CR_competitive code	Gabor filter	6	1	1	Winner-take-all	Euclidean distance	[18]	2016
	LMTrP	Gabor filter	6	1	1	Winner-take-all	Euclidean distance	[41]	2017
	ALDC	Gabor filter	6	1	1	Winner-take-all	Chi-square distance	[42]	2019
	DDBC	Gabor filter	12	1	1	Feature learning	Chi-square distance	[43]	2019
	LDDBP	Gabor filter	12	1	2	Twelve direction filtering	Chi-square distance	[44]	2020
Correlation filter	CDR	MFRAT	12	3	3	Winner-take-all	Peak-to-sidelobe ratio	[19]	2017

ive and robust competitive code (DRCC), which extracts not only the dominant direction code but also the side code of the dominant direction code by comparing two nearest neighbor responses. In 2019, Ma et al.[38] proposed the method of multi-orientation and multi-scale features discriminant learning (MOSDL). MOSDL uses a Gabor filter bank with 4 scales and 6 directions to filter the palmprint ROI images and then uses a discriminant learning strategy to select the optimal direction. Different from the work of other researchers, Leng et al.[39] paid attention to the downsampling problem in the direction coding method. They proposed a democratic voting downsampling method (DVDM) to improve the robustness and accuracy of coding-based methods. Later, they proposed extreme downsampling method (EDM)[40], in which the extreme response pixel in each local block is selected as the representative of this local block.

Jia et al.[16] proposed the first directional histogram-based method named HOL, which is a variant of the histogram of oriented gradients (HOG). HOL exploits line-shape filters or tools such as the real part of the Gabor filter and modified finite Radon transform (MFRAT) to extract the line responses and orientation of pixels instead of the gradient. Later, Luo et al.[17] proposed the local line directional pattern (LLDP) method, in which both the index numbers of the minimum line response and the maximum line response are utilized for coding. Zhang et al.[18] proposed the CR-competitive code (CR-CompC) method. CR-CompC first computes the CompC map of the palmprint by using a set of Gabor filters, then uses blockwise histogram statistics of CR-CompC as features, and uses collaborative representation-based classification with regularized least square (CRC_RLS[45]) as the classifier. Using line shape-based filters such as the Gabor filter and MFRAT and considering quadrant directions as well as thickness, Li and Kim[41] extended the local tetra pattern (LTrP) to the local microstructure tetra pattern (LMTrP).

Fei et al.[42] proposed the method of apparent and latent direction code (ALDC), which is a double-layer direction method for palmprint representation and recognition. ALDC extracts not only the apparent direction from the palmprint surface layer but also the latent direction from the energy map layer of the apparent direction. Two simple and effective schemes are used to combine the apparent and latent directional features forming a histogram feature descriptor for palmprint matching. Fei et al.[43] proposed a discriminant direction binary code (DDBC)-based palmprint descriptor. DDBC first computes the convolution difference vector (CDV) for each palmprint image. Then, DDBC learns mapping functions to project CDV into discriminant direction binary codes. Finally, the block-wise histograms of DDBC are concatenated into a discriminant direction binary palmprint descriptor. Fei et al.[44] proposed a Gaussian fusion model (EGM) to characterize the essential discriminability of different directions of palmprints and then proposed a local direction binary pattern (LDDBP) for discriminant directional feature extraction. LDDBP can better describe the direction changes and implicitly denotes the multiple dominant directional features of a palmprint. Guided by the EGM, the top three discriminant directional features are exploited from the LDDBP.

Jia et al.[19] proposed the method of complete direction representation (CDR), which uses a correlation filter, i.e., band-limited phase-only correlation (BLPOC) to extract frequency domain features from the direction representation of palmprints.

2.2 Brief introduction to the E-BOCV method

From Section 2.1, we can see that many methods based on directional features have been proposed, but most of them have not discussed the stability of directional features. Only one method, that is, the E-BOCV method proposed by Zhang et al.[22], takes into account the stability of palmprint directional features, and improves the BOCV method by using fragile bits masking strategy proposed for iris recognition[46].

In E-BOCV, fragile bits in BOCV are first extracted. Given a palmprint ROI image, six Gabor filters at the six directions are used to filter it to obtain six response maps. By binarizing each response map, the BOCV bit-plane can be obtained. E-BOCV sorts magnitude values in each response map to identify some of the smallest ones. E-BOCV regards the corresponding bits binarized from these smallest magnitudes as fragile. Then, the fragility mask is stored in a separate matrix. Consistent bits are represented as ones, while fragile bits are marked as zeros in the fragility mask. In the matching stage, when computing the Hamming distance, the fragile bits are masked. In E-BOCV, Zhang et al.[22] found that further useful information could be extracted from fragile-bit patterns. Thus, Zhang et al.[22] presented a metric fragile-bit pattern distance (FPD) to quantitatively measure the dissimilarity of two fragile-bit patterns. By fusing the modified Hamming distance and the FPD together, the original BOCV is extended to E-BOCV.

3 Methodology

For the directional feature of palmprint, after the ROI image is filtered by the filter bank, the directional value of each pixel is usually related to the direction of the maximum response. Therefore, under the guidance of the maximum response value, we first judge whether the directional value of each pixel is stable. As shown in Fig. 2, in different recognition methods, after using the filter bank to filter the ROI image, we add the DRSM module to judge the stability of each pixel′s directional response.

If the direction of a pixel is judged to be stable by the DRSM module, then the direction is encoded in a normal way. If the direction of a pixel is judged to be unstable by the DRSM module, we make a special coding of the direction of this pixel. Because of the different coding modes of different recognition methods, the special coding modes of pixels judged as unstable in direction are also different. That is, the special coding is determined by each method. Therefore, our method is a general solution to the problem of feature instability in coding methods based on direction information. In this paper, we will apply the proposed DRSM module to several classical methods based on directional information, including CompC[3], Fast CompC[19], DOC[15], LLDP[25], DRCC[20], CR_CompC[26], and BOCV[10]. When we insert the DRSM module into different recognition methods and make special coding of the direction of those unstable pixels, the modification of these recognition algorithms is very small.

3.1 The mechanism of directional response stability measurement

We use the real part of a set of Gabor filters in the K directions to filter the palmprint ROI image. It should be noted that these Gabor filters are turned to zero direct current (DC). Because the palmprint lines are dark lines, the Gabor response values along the palmprint line are negative. For the convenience of observing and displaying the filtering results, we add a negative sign to all response values so that the response values along the palm line become positive. For each pixel, we can obtain its K response values $(R_1, R_2, \cdots, R_K)$. These response values $(R_1, R_2, \cdots, R_K)$ may have three situations: 1) All of them are positive; 2) All of them are negative; 3) Part of them are positive and part of them are negative. According to the maximum response, we design two measures, i.e., direction stability response measurement 1 (DRSM_1) and direction stability response measurement 2 (DRSM_2), to address these three different situations. We sort these response values $(R_1, R_2, \cdots, R_K)$ in ascending order to obtain a new sorted value sequence $(SR_1, SR_2, \cdots, SR_K)$. Obviously, SR_K is the maximum response value, and SR_1 is the minimum response value.

The processing of the DRSM judgment mechanism is presented in Algorithm 1. For each pixel, we take $(SR_1, SR_2, \cdots, SR_K)$ as the input of the DRSM module and set $flag = 0$, where $flag$ is a logical variable. k_1, k_2, k_3 are three input parameters, which can be obtained by experiments.

If the values of $(SR_1, SR_2, \cdots, SR_K)$ are all positive (Situation (A) in Algorithm 1), we use the DRSM_1 measure to judge whether the directional response of this pixel is stable. For a pixel, if the DRSM_1 module judges that its directional response is stable, we set $flag$=1; otherwise, we set $flag$=0. DRSM_1 is defined by (1), which includes two components. The first component is the distance between T_1 and SR_1, that is, $T_1 - SR_1$, where T_1 is a value related to the maximum response SR_K. We present the calculation procedure of T_1 in Algorithm 1. The second component is $\sum_{i=2}^{N_1}(SR_K - SR_i)$, where N_1 is a parameter and $N_1 \in [2, \cdots, K]$. DRSM_1 is defined as follows:

$$DRSM\_1(T_1, N_1) = \mathrm{sgn}\left(\left(\sum_{i=2}^{N_1}(SR_K - SR_i)\right) - (T_1 - SR_1)\right). \quad (1)$$

In (1), the sign function $\mathrm{sgn}(x)$ is a special function that returns 1 for all $x > 0$ and − 1 for all $x < 0$. For $x = 0$, the value of the sign function is just zero.

The implementation of DRSM_1 can be found in part (A) of Algorithm 1. The value of T_1 is actually the value of SR_K. Thus, the value of $(T_1 - SR_1)$ equals $(SR_K - SR_1)$, which is the difference between the maximum response value SR_K and the minimum response value SR_1. The meaning of DRSM_1 is that if the directional response of a pixel is stable, the difference between the maximum response value and the minimum response value $(SR_K - SR_1)$ should be less than that of $\sum_{i=2}^{N_1}(SR_K - SR_i)$, where N_1 is k_1. Therefore, the probability that the maximum response direction flows in other directions is reduced.

Algorithm 1. DRSM judgment mechanism

Input: $(SR_1, SR_2, \cdots, SR_K)$, $flag = 0$; parameters k_1, k_2, k_3

Output: $flag$

(A): If SR_1 is bigger than 0: // The responses are all positive.

$T_1 = SR_K$

$N_1 = k_1$

If **DRSM_1** (T_1, N_1) equals 1: $flag = 1$

(B): If SR_K is less than or equal to 0://The responses are all negative.

$\mathrm{SR}' = \mathrm{SR} + (-1) \times SR_1$

$T_2 = SR'_K$

$N_2 = k_2$

If **DRSM_2** (T_2, N_2) equals 1: $flag = 1$

(C): If SR_1 is less than 0 and SR_K is bigger than 0: // Part of responses are negative.

$\mathrm{Val} = SR_1$ //Val is a temporary variable

$\mathrm{SR}' = \mathrm{SR} + (-1) \times \mathrm{Val}$

$T_1 = SR'_K + (k_1 - 1) \times (-1) \times \mathrm{Val}$

$T_2 = SR'_K + (k_3 - 1) \times (-1) \times \mathrm{Val}$

$N_1 = k_1$

$N_2 = k_3$

If (**DRSM_1** (T_1, N_1) equals 1) or (**DRSM_2**(T_2, N_2) equals 1): $flag = 1$

RETURN $flag$

If the values of $(SR_1, SR_2, \cdots, SR_K)$ are all negative (situation (B) in Algorithm 1), we add $-SR_1$ to each

value of $(SR_1, SR_2, \cdots, SR_K)$ to ensure that each modified value $(SR'_1, SR'_2, \cdots, SR'_K)$ is zero or a positive value. Then, we take $(SR'_1, SR'_2, \cdots, SR'_K)$ as the input of DRSM_2. This implementation can be found in part (B) of Algorithm 1. For a pixel, if the DRSM_2 module judges that its directional response is stable, we set $flag$=1; otherwise, we set $flag$=0. DRSM_2 is defined as follows:

$$DRSM\_2\,(T_2, N_2) = \operatorname{sgn}\left(T_2 - \sum_{i=1}^{N_2} SR'_i\right) \tag{2}$$

where N_2 is a positive integer parameter and $N_2 \in [2, \cdots, K]$. T_2 is a value related to the maximum response SR_K. Part (B) of Algorithm 1 presents the calculation procedure of T_2. Here, we add $(-1) \times SR_1$ to all the values of $(SR_1, SR_2, \cdots, SR_K)$. In this way, all the values of $(SR'_1, SR'_2, \cdots, SR'_K)$ are zero or positive. When all the Gabor filter response values of a pixel are negative, the stability of the directional response of this pixel is poor. Therefore, when the value of $(SR_1, SR_2, \cdots, SR_K)$ is updated to a positive value, we do not use DRSM_1 to judge the stability of the pixel orientation response but use more strict DRSM_2 to judge. Part (B) of Algorithm 1 shows that T_2 is the maximum value SR'_K in the value sequence of $(SR'_1, SR'_2, \cdots, SR'_K)$. In DRSM_2, when the maximum value SR'_K is greater than the sum of N_2 minimum values, i.e., $\sum_{i=1}^{N_2} SR'_i$, DRSM_2 judges that the directional response of this pixel is stable; otherwise, it is unstable.

If some response values are positive and others are negative (situation (C) in Algorithm 1), we perform DRSM_1 and DRSM_2 on this group of responses through the logical OR operation. The processing of situation (C) can be regarded as a combination of the processing of situation (A) and situation (B). We first turn this set of responses into all positive ones. That is, we add $-SR_1$ to each value of $(SR_1, SR_2, \cdots, SR_K)$ to ensure that each modified value $(SR'_1, SR'_2, \cdots, SR'_K)$ is zero or a positive value. To execute DRSM_1 and DRSM_2, we add $k_1 - 1$ times Val (Val is temporary, whose value is SR_1, as shown in Algorithm 1) and $k_3 - 1$ times Val to T_1 and T_2, respectively, for two reasons:

Reason 1. It can be seen that Val is added to each response, so if the original maximum value is directly used to compare with these updated responses, the judgment rules will be too strict, so that a large number of sampling pixels will be mistaken for unstable pixels, leading to the adoption of the same coding rules, which makes the feature vectors lack specificity. Therefore, we add an additional val of $k_1 - 1$ times and a val of $k_3 - 1$ times to T_1 and T_2. For the strict and loose design of rules, see Section 5 later.

Reason 2. A new parameter k_3 has been added to situation (C). From the most direct and understandable point of view, situation (C) should actually introduce two new parameters k_3 and k_4 because there are T_1 and T_2 to be set. If this is done, the solution space will be very large because k_i belongs to 1–K, where K is the number of directional responses, which will lead to more experiments to determine a set of suitable k configurations.

So why is k_1 used for T_1 and k_3 used for T_2 instead of k_1 for T_1 and k_2 for T_2? The reason is that k_1 is associated with situation (A), and situation (A) deals with the case of all positive responses. We think that when the responses are all positive, the probability of this group of responses being stable is greater. When Val is added to this group of responses, it becomes all positive, so using k_1 for T_1 can control this part of the rules to be less strict. For DRSM_2, because this set of responses is positive and negative, it cannot be as strict as when dealing with all negative responses. If k_2 is directly used as situation (C), it is easy to make the sampling points fail to pass the filtering rules, so it is more convenient to adjust the filtering strength of the rules by using an extra parameter for DRSM_2. Both filtering rules and parameter settings can adjust the filtering strength of sampling points with a certain kind of response, so we combine them to enhance the flexibility of the framework. An identification flag is output for each sampling point through the DRSM judgment mechanism. If $flag = 1$, select the direction according to the rules of the original method, and code normally. If $flag = 0$, DRSM coding is performed without selecting direction information.

The values of N_1 and N_2 will be determined by three hyperparameters k_1, k_2 and k_3.

3.2 Insert DRSM mechanism into different directional feature-based methods

We insert the proposed DRSM mechanism into seven classical methods based on directional response, including CompC[7], Fast CompC[32], DOC[9], LLDP[12], DRCC[10], CR_CompC[13], and BOCV[8]. For different methods, the insertion mode of the DRSM mechanism is also different. In this subsection, we introduce the insertion modes one by one. In addition, when introducing different methods, we use the formulas and symbols in the original paper so that readers can better understand these methods.

For different methods, to encode the features of the pixels whose directional responses are judged to be unstable by DRSM as a part of the final feature vector, which can be applied to the corresponding feature matching methods, we adopt the same encoding form (feature form) as the original method. In feature matching, we realize the distinction on the feature level, that is, the feature similarity between any two sampling points. The distance between DRSM coding and any normal coding should be as same as possible.

3.2.1 Insert DRSM mechanism into CompC

The method of CompC[12] uses the real part of the Gabor filter in six directions ($j\pi/6,\ j=\{0,1,\cdots,5\}$) to convolve the palmprint image and then obtain six response values. For each pixel at the position (x,y), the winner-take-all rule is used to determine the direction of this pixel, that is, the index number of the maximum response.

$$\arg\ \min_{j}\left(I\left(x,y\right)\times\Psi_R(\,x,y,\omega,\theta_j\,)\right) \tag{3}$$

where I is the palmprint ROI image and Ψ_R denotes the real part of the Gabor filter.

The direction value of each pixel is between 0 and 5, which can be represented by the three bits listed in Table 2. Let P and Q be two CompCs, and their distance D can be measured by the Hamming distance.

$$D\left(P,Q\right)=\frac{\sum_{y=0}^{N}\sum_{x=0}^{N}\sum_{i=1}^{3}\left(P_i(\,x,y)\otimes Q_i(x,y)\right)}{3N^2} \tag{4}$$

where P_i and Q_i are the i-th bit planes of P and Q, respectively; $\otimes$ is the bitwise XOR operation; N is the size of the palmprint image.

Table 2 3 bit coding method for six directions

Orientation	Bit 1	Bit 2	Bit 3
0	0	0	0
1	0	0	1
2	0	1	1
3	1	1	1
4	1	1	0
5	1	0	0

To insert the DRSM mechanism into the CompC, the DRSM mechanism is used to judge whether the directional response of each pixel is stable in the process of obtaining directional features by the CompC. If the DRSM mechanism judges that the directional response of a pixel is unstable, we set $flag$=0 on this pixel and define the direction of this pixel as the seventh direction, which is also called the code of DRSM_CompC. The coding of DRSM_CompC is defined as follows:

$$[-1,-1,-1]\,. \tag{5}$$

To make the distance from the DRSM_CompC code to any CompC in Table 2 equal, we modify the similarity measurement (4) as follows:

$$D\left(P,Q\right)=\sum_{y=0}^{N}\sum_{x=0}^{N}\sum_{i=1}^{3}\operatorname{sgn}\left(\operatorname{abs}\left(P_i\left(x,y\right)-Q_i\left(x,y\right)\right)\right) \tag{6}$$

where P_i and Q_i are the i-th bit planes of P and Q, respectively; abs () is the absolute value operation function; sgn () is a symbolic function.

It is worth noting that this modification will not affect the experimental results of the original method, just to be able to calculate the distance between DRSM_CompC code and other CompCs. Moreover, after modification, it is more convenient to realize and faster to calculate the distance measure.

Under this setting, we set the distance between DRSM_CompC code and other CompC codes to 3. Of course, this distance can also be adjusted to a positive integer of other values, such as 1, 2, 4, etc., but to retain a certain degree of discrimination. This code will not occupy a large proportion in the whole feature vector. The final experimental results show that setting the distance to 3 can achieve better results. Through the above settings, DRSM can be successfully inserted into the CompC.

3.2.2 Insert the DRSM mechanism into BOCV

In CompC, only the minimal response direction feature is used according to the winner-take-all rule. Guo et al.[13] thought that if only one direction feature is used to represent local features, some structural information may be lost. Therefore, they proposed the BOCV method. BOCV binarizes the Gabor responses in six directions by introducing the threshold T, so each sampling point position obtains the 6 bit feature, thus retaining all the direction information to the greatest extent.

$$P_j^b\left(x,y\right)=\begin{cases}1, & \text{if } R_j\left(x,y\right)<T_j\\ 0, & \text{otherwise}\end{cases} \tag{7}$$

where R_j represents the response in the j direction, $T_j=0, j=\{0,1,2,3,4,5\}$.

BOCV uses the Hamming distance to calculate similarity:

$$D\left(\mathrm{P^b},\mathrm{Q^b}\right)=\frac{\sum_{y=1}^{\mathrm{M}}\sum_{x=1}^{\mathrm{N}}\sum_{\mathrm{j}=0}^{5}\left(P_j^b\left(x,y\right)\otimes Q_j^b\left(x,y\right)\right)}{6\times N^2} \tag{8}$$

where P^b and Q^b are the bit planes of I_P and I_Q, $\otimes$ is the bitwise XOR operation, and N is the size of the palmprint image.

If the DRSM mechanism judges that the directional response of a pixel is unstable, then we set $flag=0$ on this pixel and define the code of DRSM_BOCV as follows:

$$[-1,-1,-1,-1,-1,-1]\,. \tag{9}$$

We calculate the distance according to the distance measurement of DRSM_CompC. Under this setting, the distance between DRSM_BOCV code and any BOCV code is 6. Similar to the operation of DRSM_CompC, this distance can also be set to any integer, and only a simple

judgment is needed in the calculation. Because this is not the focus of this paper, we have not performed many experiments to find the best distance value. From the experimental results, when the distance is set to 6, the performance can still be improved.

3.2.3 Insert the DRSM mechanism into DOC

Double-orientation code (DOC)[14] uses Gabor filters in n_θ directions to filter palmprint ROI images and selects two directional index values of the smallest response for coding by the winner-take-all rule.

$$[O_p(x,y), O_s(x,y)] = \arg \min_{j1,j2} R_j(x,y), \quad j = \{0,1,\cdots,n_\theta - 1\}. \tag{10}$$

To improve the recognition rate, Fei et al.[14] proposed a nonlinear angle matching score method to evaluate the similarity of DOC codes. Before calculating the similarity score of two positions, some variables need to be defined. $code_{dis_{\alpha\beta}}$ is the distance between two single-orientation codes. Assuming that A and B represent sampling points at two locations, $code_{dis_{\alpha\beta}}$ can be calculated by the following formula:

$$code_{dis_{\alpha\beta}} = \min\left(\left|{O^A}_\alpha - {O^B}_\beta\right|, n_\theta - \left|{O^A}_\alpha - {O^B}_\beta\right|\right). \tag{11}$$

P_{1score} and P_{2score} can be calculated by the following formulas:

$$P_{1score} = ori_{score}\left(code_{dis_{pp}}\right) + ori_{score}\left(code_{dis_{ss}}\right) \tag{12}$$

$$P_{2score} = ori_{score}\left(code_{dis_{ps}}\right) + ori_{score}\left(code_{dis_{sp}}\right) \tag{13}$$

where

$$ori_{score(code_{dis})} = \frac{1}{e^{k \times code_{dis}}} \tag{14}$$

and k is a hyperparameter, which is related to the Gabor filter with n_θ directions. In the original text, the best experimental results are obtained when $n_\theta = 6$ and $k = 1.6$ are set, so the same configuration as the original text is adopted in this experiment.

Finally, the DOC similarity between position A and position B is calculated as follows:

$$P_{score(A,B)} = \max\left(P_{1score(A,B)}, P_{2score(A,B)}\right). \tag{15}$$

To be more compatible with the distance calculation formula of DOC and meet the unique coding principle as much as possible, we set DRSM_DOC coding as follows:

$$[O_p(x,y), O_s(x,y)] = [-1,-1]. \tag{16}$$

Due to the introduction of DRSM_DOC code, when calculating the distance, there are the following three situations in the distance calculation of any two positions:

1) If the two codes are both positive, they are calculated and matched in the original way.

2) If one of the two codes is positive and the other is negative, we make the following definition:

$$code_{dis_{pp}}, code_{dis_{ps}}, code_{dis_{sp}}, code_{dis_{ss}} = 3. \tag{17}$$

3) If both codes are all negative, the following definition is made:

$$code_{dis_{pp}}, code_{dis_{ps}}, code_{dis_{sp}}, code_{dis_{ss}} = 0. \tag{18}$$

To ensure that the distance between DRSM_DOC code and any DOC code is equal, we set the distance between DRSM_DOC code and any DOC code to 3. When calculating the distance, we only need to add simple logic judgment to the original matching formula.

3.2.4 Insert the DRSM mechanism into DRCC

Discriminative and robust competitive code (DRCC)[15] extracts the direction information of palmprint images by using a set of 6-direction circular Gabor filters. Because there may be a problem of poor discrimination when using the direction of the maximum response alone, DRCC adopts the direction of the maximum response and the relationship between its left and right adjacent directions as the object of subsequent coding and uses a Gaussian template to weight the positioning information of adjacent areas to improve the accuracy and stability of the positioning code with advantages of discrimination and robustness.

According to the winner-take-all rule, the maximum response direction of the (x,y) position is

$$C(x,y) = \arg \max_j R_j(x,y), j = \{0,1,\cdots,5\}. \tag{19}$$

The adjacent direction setting rule of the maximum response direction and the coding mode of the relationship between its responses are obtained from (20)–(22), respectively, where C is the direction of the maximum response.

$$C_{left} = \begin{cases} C+1, & \text{if } 0 \le C \le 4 \\ 0, & \text{if } C = 5 \end{cases} \tag{20}$$

$$C_{right} = \begin{cases} C-1, & \text{if } 1 \le C \le 5 \\ 5, & \text{if } C = 0 \end{cases} \tag{21}$$

$$\widetilde{C}_s = \begin{cases} 1, & \text{if } R_{C_{left}}(x,y) \ge R_{C_{right}}(x,y) \\ 0, & \text{otherwise.} \end{cases} \tag{22}$$

Therefore, the DRCC code $[C, \widetilde{C}_s]$ is obtained, and for-

mula (23) is used to determine the similarity between the two palmprint images.

$$M(X,Y) = \frac{1}{2N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left[\left(\widetilde{C_X}(i,j)\cap\widetilde{C_Y}(i,j)\right)\right]+ \left(\left(\widetilde{C_X}(i,j)\cap\widetilde{C_Y}(i,j)\right)\right)\cap \neg\left(\left(\widetilde{C_{sX}}(i,j)\oplus\widetilde{C_{sY}}(i,j)\right)\right) \quad (23)$$

where $\widetilde{C_X}$, $\widetilde{C_{sX}}$ and $\widetilde{C_{sY}}$ and $\widetilde{C_{sY}}$ represent the maximum response direction coding plane and the adjacent relation coding plane of the input images X and Y, respectively. N is the image size, $\cap$ represents the AND operation and $\oplus$ represents the XOR operation.

According to its encoding method, we set the DRSM_DRCC encoding with $flag = 0$.

$$\left[C, \widetilde{C}_s\right] = [-1, -1]. \quad (24)$$

When calculating the distance, (23) can be equivalently converted into:

$$temp = \widetilde{C_X} \cap \widetilde{C_Y} \quad (25)$$

$$score_{max} = \sum_{i=1}^{N}\sum_{j=1}^{N} temp(x,y) \quad (26)$$

$$score_{neib} = \sum_{i=1}^{N}\sum_{j=1}^{N} temp(x,y) \times \left(\widetilde{C_{sX}}(x,y)\cap\widetilde{C_{sY}}(x,y)\right) \quad (27)$$

$$score = \frac{score_{max} + score_{neib}}{2\times N^2} \quad (28)$$

where $\cap$ represents the AND operation, and N represents the size of the image.

3.2.5 Insert the DRSM mechanism into Fast CompC

Fast CompC convolves the palmprint image with two orthogonal Gabor filter templates. Each position has two directional responses, and then obtains the logical relationship between them.

$$bit = R_j(x,y) > R_{j+90^\circ}(x,y) \quad (29)$$

where j can be 0°, 30°, or 60°.

Therefore, each position is represented by one bit feature. To ensure the same distance between DRSM_ Fast CompC coding and any Fast CompC, similar to the operation in competitive coding, we set it to [–1].

We calculate the distance according to the modified distance measurement (6).

3.2.6 Insert the DRSM mechanism into the LLDP

Local line directional pattern (LLDP) first convolves the palmprint ROI image by using the real part of the circular Gabor filter with 12 directions and obtains the response value of 12 directions at each position. By encoding the maximum response direction and the minimum response direction, the LLDP code is finally generated:

$$LLDP\ code = (t_1 - 1)\times 12^1 + (t_2 - 1)\times 12^0 \quad (30)$$

where t_1 is the direction of the maximum response and t_2 is the direction of the minimum response.

Then, the whole coded image is divided into nonoverlapping small areas $\{A^1, \cdots, A^N\}$, and the histogram is extracted from each area.

LLDP adopts a set of 2D Gabor filters with 12 directions for convolution, so the maximum subscript is 12, and the minimum subscript is 1, so the range of C is 0–143. That is, each sampling point will be assigned a number between 0 and 143. LLDP adopts local histogram statistics. Therefore, to realize the uniqueness of the DRSM_LLDP code, according to its matching mode, the DRSM_LLDP code of this position is set to 144 as an additional feature dimension so that $t_1 = 12$, $t_2 = 13$.

After adding DRSM_LLDP coding, the coding range of each position is 0–144. Therefore, the DRSM module can be easily inserted into the LLDP method by simply modifying the range of histogram statistics. Notably, when the DRSM_LLDP coding is processed, it does not fully conform to the DRSM coding principle because of the histogram statistics.

3.2.7 Insert the DRSM mechanism into CR_CompC

The CR_CompC method is the histogram-based variant of CompC. First, CompC is run on the input image to obtain the competitive coding map of the input image. Then, the map is divided into several regions, and the number of times in each region in all directions is counted so that the histogram features of each region are obtained, which are expressed as h_s, where S represents the subscript of the region, and then all of them are connected to obtain the CR_CompC representation of the input image, where N is the number of regions.

$$h = [h_1, h_2, \cdots, h_N]. \quad (31)$$

Then, the CRC_RLS model is adopted to classify the CR_CompC features of the image.

The process of obtaining feature vectors by CR_CompC is mainly related to the execution process of competitive coding. In the original competition code, each position will be assigned a direction j, where $j = \{0, \cdots, 5\}$. To insert DRSM into CR_CompC, we replace the original CompC with DRSM_CompC so that the direction of each position is now $j = \{0, \cdots, 6\}$. From the histogram

point of view, there is only one more dimension, and other parts of this method remain unchanged.

4 Experiments

4.1 Palmprint databases used for experiments

In this paper, seven palmprint image databases are exploited for performance evaluation, including the Hong Kong Polytechnic University palmprint database II (PolyU II)[24], the blue band of the Hong Kong Polytechnic University Multispectral (PolyU M_B) palmprint database[47], the Hefei University of Technology (HFUT) palmprint database[19], the Hefei University of Technology Cross Sensor (HFUT CS) palmprint database[48], the Tongji University palmprint (TJU-P) database[18], and the IIT Delhi palmprint image database[49]. After preprocessing, the ROI subimages were cropped. The ROI size of all databases is 128 × 128. The detailed descriptions of the above databases are listed in Table 3.

4.2 Experimental results

We conduct both identification and verification experiments. In the identification experiment, the nearest neighbor rule is used for classification. The statistical value, i.e., recognition rate, is exploited to evaluate the identification performance, which is the rank 1 identification rate. We also conduct a verification experiment. The statistical value, i.e., equal error rate (EER), is used for performance evaluation. For the PolyU II, PolyU M_B, CS, HFUT, and TJU-P databases, the first three in the first stage are used as training sets, and all in the second stage are used as test sets. For the IITD database, because the number of each category is small (most of them are less than or equal to 5), we take the first one of each identity as the training set and the others as the test set.

Before the experiments, the values of the parameters of k_1, k_2 and k_3 for different databases should be determined. For each method, after many experiments on each dataset, a set of k_1, k_2 and k_3 values of each method on each dataset are obtained. We list their values in Table 4.

The performance comparisons between the original methods (CompC, BOCV, DOC, DRCC, Fast CompC, LLDP, CR_CompC) and their versions by inserting the DRSM mechanism (CompC + DRSM, BOCV + DRSM, DOC + DRSM, DRCC + DRSM, Fast CompC + DRSM, LLDP + DRSM, CR_CompC + DRSM) are listed in Tables 5–11, respectively.

In the original paper of the Fast CompC method, the author of the paper did not specify the value of j. To determine the direction of j in the subsequent experiments, we conducted experiments on the HFUT CS database and chose the direction j with the best performance as the final experimental setting. The experimental results are shown in Table 12. As shown in Table 12, when the value of j is 30°, the recognition performance of the Fast CompC method is the best. Therefore, in the Fast Com-

Table 3 The details of six palmprint databases

Database	Type	Touch?	Individual number	Palm number	Session number	Session interval	Image number of each palm	Total image number
PolyU Ⅱ	2D Palmprint	Yes	193	386	2	2 months	10×2	7 752
PolyU M_B	2D Palmprint	Yes	250	500	2	9 days	6×2	6 000
HFUT	2D Palmprint	Yes	400	800	2	10 days	10×2	16 000
CS	2D Palmprint	No	100	200	2	10 days	10×2×3	12 000
TJU-P	2D Palmprint	No	300	600	2	61 days	10×2	12 000
IITD	2D Palmprint	No	230	460	1	N/A	5 – 6	2 601

Table 4 k_1, k_2, k_3 experimental configurations of seven methods on six databases

Methods	PolyU Ⅱ	PolyU M_B	CS	TJU-P	HFUT	IITD
CompC	$k_1=3, k_2=3, k_3=3$	$k_1=3, k_2=3, k_3=3$	$k_1=2, k_2=2, k_3=3$	$k_1=2, k_2=4, k_3=3$	$k_1=4, k_2=5, k_3=3$	$k_1=3, k_2=5, k_3=3$
LLDP	$k_1=6, k_2=9, k_3=5$	$k_1=6, k_2=5, k_3=9$	$k_1=9, k_2=8, k_3=10$	$k_1=7, k_2=7, k_3=5$	$k_1=5, k_2=5, k_3=5$	$k_1=5, k_2=5, k_3=5$
DOC	$k_1=3, k_2=5, k_3=4$	$k_1=4, k_2=4, k_3=4$	$k_1=4, k_2=4, k_3=5$	$k_1=2, k_2=5, k_3=3$	$k_1=2, k_2=5, k_3=2$	$k_1=5, k_2=5, k_3=3$
DRCC	$k_1=5, k_2=4, k_3=3$	$k_1=5, k_2=5, k_3=2$	$k_1=4, k_2=4, k_3=3$	$k_1=4, k_2=5, k_3=3$	$k_1=4, k_2=5, k_3=3$	$k_1=5, k_2=5, k_3=4$
Fast CompC	$k_1=2, k_2=5, k_3=2$	$k_1=2, k_2=5, k_3=2$	$k_1=2, k_2=5, k_3=2$	$k_1=2, k_2=5, k_3=2$	$k_1=2, k_2=5, k_3=2$	$k_1=2, k_2=5, k_3=3$
CR_CompC	$k_1=3, k_2=4, k_3=3$	$k_1=5, k_2=5, k_3=3$	$k_1=3, k_2=4, k_3=3$	$k_1=3, k_2=5, k_3=5$	$k_1=3, k_2=5, k_3=3$	$k_1=5, k_2=5, k_3=2$
BOCV	$k_1=3, k_2=5, k_3=3$	$k_1=3, k_2=4, k_3=3$	$k_1=2, k_2=2, k_3=5$	$k_1=4, k_2=2, k_3=3$	$k_1=2, k_2=4, k_3=3$	$k_1=2, k_2=5, k_3=5$

Table 5 Performance comparison between CompC and CompC + DRSM

	Recognition rate (%)		EER (%)	
	CompC	CompC + DRSM	CompC	CompC + DRSM
PolyU II	100	**100**	0.005 3	**0.001 7**
PolyU M_B	100	**100**	0.000 233 8	**0**
HFUT CS	99.58	**99.68**	0.52	**0.41**
TJU-P	100	**100**	0.14	**0.066 8**
HFUT	99.88	**99.93**	0.21	**0.18**
IITD	91.45	**91.97**	5.25	**5.22**

Table 6 Performance comparison between BOCV and BOCV + DRSM

	Recognition rate (%)		EER (%)	
	BOCV	BOCV + DRSM	BOCV	BOCV + DRSM
PolyU II	100	**100**	0.000 26	**0.000 1**
PolyU M_B	100	**100**	0	**0**
HFUT CS	99.90	**99.93**	0.23	**0.21**
TJU-P	100	**100**	**0.006 8**	0.006 9
HFUT	99.90	**99.93**	0.10	**0.081 7**
IITD	92.25	**92.34**	5.23	**5.23**

Table 7 Performance comparison between DOC and DOC + DRSM

	Recognition rate (%)		EER (%)	
	DOC	DOC + DRSM	DOC	DOC + DRSM
PolyU II	98.48	**98.82**	0.80	**0.70**
PolyU M_B	98.00	**98.37**	1.10	**0.87**
HFUT CS	88.85	**89.75**	5.68	**5.38**
TJU-P	95.22	**95.80**	2.75	**2.42**
HFUT	94.64	**95.53**	2.43	**2.11**
IITD	65.3	**67.07**	18.60	**18.00**

Table 8 Performance comparison between DRCC and DRCC + DRSM

	Recognition rate (%)		EER (%)	
	DRCC	DRCC + DRSM	DRCC	DRCC + DRSM
PolyU II	98.74	**98.92**	0.76	**0.69**
PolyU M_B	98.27	**98.47**	0.93	**0.87**
HFUT CS	89.58	**90.03**	5.16	**4.90**
TJU-P	95.57	**95.80**	2.44	**2.17**
HFUT	95.05	**95.63**	2.16	**1.92**
IITD	65.95	**67.07**	17.8	**17.38**

pC method experiment, we fixed the value of j to 30°.

From Tables 5–11, it can be seen that the recognition performances of seven recognition methods on six palmprint databases are all improved after inserting the DRSM mechanism. The experimental results show the effectiveness of our proposed method.

Table 9 Performance comparison between Fast CompC and Fast CompC + DRSM

	Recognition rate (%)		EER (%)	
	Fast CompC	Fast CompC + DRSM	Fast CompC	Fast CompC + DRSM
PolyU II	99.95	**99.97**	0.13	**0.078 8**
PolyU M_B	99.97	**99.97**	0.23	**0.077 9**
HFUT CS	98.98	**99**	1.02	**1.01**
TJU-P	99.28	**99.67**	0.63	**0.44**
HFUT	99.23	**99.60**	0.88	**0.48**
IITD	87.30	**88.74**	7.06	**6.77**

Table 10 The performance comparison between the LLDP and LLDP + DRSM

	Recognition rate (%)		EER (%)	
	LLDP	LLDP + DRSM	LLDP	LLDP + DRSM
PolyU II	100	**100**	0.051 7	**0.027 4**
PolyU M_B	100	**100**	0.002 6	**0.001 2**
HFUT CS	99.58	**99.68**	1.48	**1.36**
TJU-P	100	**100**	0.57	**0.32**
HFUT	99.88	**99.93**	0.34	**0.20**
IITD	91.45	**94.26**	5.52	**4.03**

Table 11 Performance comparison between CR_CompC and CR_CompC + DRSM

	Recognition rate (%)		EER (%)	
	CR_ CompC	CR_CompC + DRSM	CR_ CompC	CR_CompC + DRSM
PolyU II	99.74	**99.82**	0.19	**0.18**
PolyU M_B	99.47	**99.67**	0.17	**0.13**
HFUT CS	95.08	**95.63**	1.89	**1.66**
TJU-P	98.68	**99.15**	0.54	**0.40**
HFUT	98.38	**98.61**	0.64	**0.53**
IITD	82.30	**83.79**	7.2	**6.45**

Table 12 The recognition result of Fast CompC for different j on the HFUT CS database

	Recognition rate (%)	EER(%)
$j = 0°$	98.10	1.40
$j = 30°$	98.98	1.02
$j = 90°$	98.60	1.49

4.3 Running time

All the experiments are carried out on Intel(R) i5-10300H (3.2 GHz) quad-core PC and Windows 10 operating system using MATLAB 2021b. It can be found from Table 13 that due to the simplicity of DRSM design, the introduction of the DRSM module into these methods only brings a small extra time overhead in feature extraction and hardly brings burden to the original method in feature matching. Therefore, compared with the benefits of the DRSM module, this time cost is worthwhile. We believe that the efficiency and simplicity of this method will be conducive to further research and application in the future.

Table 13 Running time (ms) of different methods

Methods	Feature extraction	Matching	Total
CompC	15.70	0.210 0	15.910 0
CompC + DRSM	17.40	0.211 0	17.611 0
LLDP	36.01	0.074 6	36.084 6
LLDP + DRSM	37.40	0.075 0	37.475 0
DOC	25.40	0.320 0	25.720 0
DOC + DRSM	26.40	0.330 0	26.730 0
DRCC	19.10	0.142 0	19.242 0
DRCC + DRSM	27.50	0.145 0	27.645 0
Fast CompC	4.90	0.095 0	4.995 00
Fast CompC + DRSM	16.60	0.095 0	16.695 0
CR_CompC	34.60	0.004 0	34.604 0
CR_CompC + DRSM	64.10	0.004 0	64.104 0
BOCV	14.60	0.723 5	15.323 5
BOCV + DRSM	16.80	0.726 0	17.526 0

4.4 Ablation study

To further study the function of each part of DRSM, we conduct experiments on the decision rules designed in three scenarios (they are simplified as rule A, rule B and rule C). All experiments are based on the DRSM_ CompC method and PolyU II databases.

4.4.1 Single rule VS. multiple rules

In DRSM, rule A, rule B and rule C are used to deal with all positive (A), all negative (B), and part positive and part negative responses (C), respectively. Table 14 shows the experimental results after the original CompC and the application of various rules alone. As mentioned above, any sampling point may contain any one of the three types of responses, and it may occur anywhere. Therefore, it is expected to reduce the damage to the performance of the original method caused by instability by processing any type of response alone. However, this does not mean that it has to be the case. As seen from the application of rule C alone, because the actual situation of palmprint images is very complicated, different acquisition environments and acquisition strategies will all affect the images, so it is a safer way to comprehensively use the three rules. And it is also hopeful to deal with the complex feature instability problem to the greatest extent.

4.4.2 Performance under different k_1, k_2, k_3

As mentioned above, the setting of parameters k_1, k_2, k_3 controls the strictness of judgment rules. To analyse the role of parameters k_1, k_2, k_3 in controlling DRSM, we show the proportion of DRSM sampling points in the total sampling points under different k_1, k_2, k_3 configurations and the experimental results in Table 15. As seen from Table 14, with the change in the V value, the proportion of DRSM points in the total sampling points is constantly changing, which also means that the strictness of filtering rules is constantly changing. From the experimental results, it can be found that the stricter the better. The stricter the filtering is, the lower the difference of original features is. Because the proportion of DRSM points is increasing, and the features of each DRSM point are the same. For items 4 and 5, why do they reduce the performance of the original method when the proportion is very small? Our understanding is that the EER of the original method is already in a small position in this database, and any feature change may bring about a large change. DRSM does not erase all the points with low responses or points with similar responses but selects some of these potential unstable points instead of setting a threshold directly. Therefore, this is a "softer" way because not all points are useless. For these points, the probability of instability is high but not certain. Therefore, for the situation of item 4 and item 5, we think that it may be that the valuable points among these potential unstable points have just been selected, thus affecting the difference of the original features, even if the proportion is very small.

Table 14 The ablation experiments for different rules

	Portion (%)	k_1, k_2, k_3	Rule A	Rule B	Rule C	Recognition rate (%)	EER (%)
Baseline						1	0.005 3
A	2.3	$k_1 = 3$	√			1	0.002 1
B	0.097	$k_2 = 5$		√		1	0.005 0
C	0	$k_2 = 4, k_3 = 3$			√	1	0.005 3
A+B+C	2.54	$k_1 = 3, k_2 = 3, k_3 = 3$	√	√	√	1	0.001 7

Table 15 The performance under different k_1, k_2, k_3 settings

	Portion (%)	k_1, k_2, k_3	Recognition rate (%)	EER (%)
Baseline			100	0.005 3
A+B+C	2.54	$k_1 = 3, k_2 = 3, k_3 = 3$	100	0.001 7
A+B+C	2.34	$k_1 = 4, k_2 = 3, k_3 = 3$	100	0.002 1
A+B+C	2.83	$k_1 = 4, k_2 = 3, k_3 = 4$	100	0.003 1
A+B+C	0.20	$k_1 = 3, k_2 = 2, k_3 = 3$	100	0.006 5
A+B+C	0.098	$k_1 = 4, k_2 = 2, k_3 = 3$	100	0.006 2

4.5 Comparison with the E-BOCV method

E-BOCV directly introduces fragile bit masking into BOCV to address the unstable bit problem in BOCV. We applied DRSM in BOCV, so we want to make a performance comparison between E-BOCV and our method, i.e., BOCV + DRSM. Table 16 lists the recognition performance comparison between E-BOCV and BOCV + DRSM. Table 17 lists the computational cost comparison between E-BOCV and BOCV + DRSM. As seen from Tables 16 and 17, although E-BOCV has achieved better recognition results on the PolyU II and IITD databases than BOCV + DRSM, it needs to consume more time and storage space because it needs to use six extra planes to store the corresponding fragile bits matrix. Second, fragile bits masking cannot be applied to other palmprint recognition methods. It can be applied to BOCV because there is a one-to-one correspondence between its coding features and responses, that is, six directional responses corresponding to six bit codes. Therefore, as long as there is no one-to-one correspondence between responses and coding features, the fragile bit masking strategy in iris recognition cannot even be applied to other methods of bit-by-bit coding features, let alone other types of features, such as histogram statistical features and other artificially designed coding features. Compared with fragile bits masking′s practice of specifying specific types of features, DRSM is independent of specific feature representations and stands at a higher level. It considers and designs a general mechanism for directional feature-based methods in palmprint recognition, which does not require additional space overhead and is more flexible and universal.

Table 16 The recognition performance comparison between E-BOCV and BOCV + DRSM

	Recognition rate (%)		EER(%)	
	E-BOCV	BOCV + DRSM	E-BOCV	BOCV + DRSM
PolyU II	100	100	**0**	0.000 1
PolyU M_B	100	100	0	0
HFUT CS	**99.95**	99.93	0.30	**0.21**
TJU-P	100	100	0.033 4	**0.006 9**
HFUT	99.93	99.93	0.14	**0.081 7**
IITD	**93.09**	92.34	**4.72**	5.23

Table 17 The computational cost (ms) comparison between E-BOCV and BOCV + DRSM

	Feature extraction	**Matching**	**Total**
BOCV + DRSM	**16.80**	**0.726 0**	**17.526 0**
E-BOCV	24.00	0.745 0	24.745 0

5 Discussions

Theoretically, there are many possible ways to implement the DRSM mechanism. For example, when calculating the total amount and distance, it decreases from the second maximum response instead of increasing from the minimum response. This seems to be more reasonable because it can highlight the strength advantage of maximum response to the greatest extent. However, there are several problems with this operation. First, this operation will lead to too strict filtering rules, so that many normal sampling points are wrongly regarded as unstable points, which will lead to a sharp increase in the number of sampling points using the same code in the total sampling points, thus greatly weakening the difference of feature vectors themselves. Second, there is a lack of control over the strictness of filtering rules. Because at the beginning we stand at a very strict level, and the adjustment of a set of parameters cannot significantly affect the already strict level. Therefore, DRSM adopts a soft method, increasing from the minimum value. However, as mentioned above, there are still many possibilities for the design of DRSM mechanisms, and more operations can be derived by the idea of DRSM coding, which also reflects the flexibility of DRSM mechanisms.

From the judgment rules of DRSM, it can be seen that for the sampling points passing through the filtering module, the relationship between the top X responses may be vague. DRSM is not designed to ensure the maximum response strength relative to the top X response. Therefore, the potential direction information may fluctuate between top X responses. Fig. 3 shows an example of an area in the top-3 direction. Perhaps this is not what we want to see because we want either the response value to be stable and the direction not to flow or $flag = 0$ and the direction to remain fixed. However, the DRSM judg-

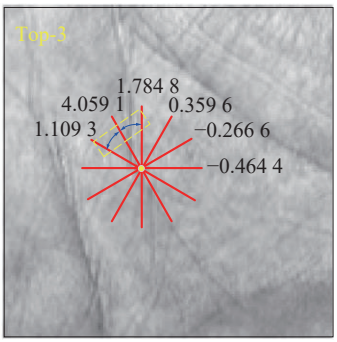

Fig. 3 Area formed by the top-3 direction response

ment module is not used to judge whether a sampling point is on the obvious palmprint line. Because the line segments in the palmprint image are very messy, it cannot be done with the help of a set of rules alone. Second, the principle of directional feature coding is to make the similarity between adjacent directions the closest, while the distance between nonadjacent directions is as large as possible. Compared with the direction information fluctuating in a random large range, it is acceptable in a controllable small range. This not only reduces the influence of feature instability to a certain extent but also prevents the filtering rules from being too strict to affect the normal sampling point features. DRSM achieves a balance between the two.

Through statistics, it is found that the proportion of DRSM points in all sampling points is very small. For example, on the PolyU II database, when $k_1 = 3$, $k_2 = 3$, $k_3 = 3$, and the number of sampling point is 32×32, it accounts for approximately 2.5% of the total. In the design of the DRSM filter module, we only focus on the maximum response instead of responding in multiple directions or even all directions. The reason for this is that the unit value of the maximum response is the largest in a group of responses. Second, if more directional response information is added to set rules, the coupling between rules will increase sharply, which is not conducive to the design of rules because these responses have mutual influence. Moreover, the rules will become increasingly complex, which will affect the speed of feature extraction. It is worth noting that all the methods we use this time use Gabor filters as feature extraction tools. We have also conducted extensive experiments on the methods of using Gaussian filters and MFART as direction extractors, but we cannot see obvious performance improvement. We think the reason is that the current DRSM design may not be suitable for methods other than Gabor filters, because there are certain shape differences among various filters, so the areas involved in the process of obtaining direction responses are different. However, this is our first attempt in the field of palmprint recognition. We believe that the problem of feature instability in palmprint recognition is widespread, our follow-up work will focus on how to design a unified scheme for all kinds of filters to improve the performance of all kinds of directional coding information.

6 Conclusions

In this paper, we proposed the DRSM for the palmprint recognition method based on directional features for the first time to address the problem of directional response instability. Our method can be successfully applied to many classical methods based on directional features and has achieved recognition performance improvement on six public databases, providing an effective solution to the problem of unstable directional responses. Finally, our research shows that although the current palmprint recognition methods have achieved very good recognition results, the problem of directional response instability still more or less weakens its best results, so it is of great significance to study the problem of directional response instability.

Acknowledgements

This work is partly supported by National Science Foundation of China (No. 62076086).

Declarations of conflict of interest

The authors declared that they have no conflicts of interest to this work.

References

[1] L. K. Fei, G. M. Lu, W. Jia, S. H. Teng, D. Zhang. Feature extraction methods for palmprint recognition: A survey and evaluation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 2, pp. 346–363, 2019. DOI: 10.1109/TSMC.2018.2795609.

[2] L. K. Fei, B. Zhang, Y. Xu, C. W. Tian, I. Rida, D. Zhang. Jointly heterogeneous palmprint discriminant feature learning. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 9, pp. 4979–4990, 2022. DOI: 10.1109/TNNLS.2021.3066381.

[3] S. A. Maadeed, X. D. Jiang, I. Rida, A. Bouridane. Palmprint identification using sparse and dense hybrid representation. *Multimedia Tools and Applications*, vol. 78, no. 5, pp. 5665–5679, 2019. DOI: 10.1007/s11042-018-5655-8.

[4] I. Rida, R. Herault, G. L. Marcialis, G. Gasso. Palmprint recognition with an efficient data driven ensemble classifier. *Pattern Recognition Letters*, vol. 126, pp. 21–30, 2019. DOI: 10.1016/j.patrec.2018.04.033.

[5] W. Jia, J. Gao, W. Xia, Y. Zhao, H. Min, J. T. Lu. A performance evaluation of classic convolutional neural networks for 2D and 3D palmprint and palm vein recognition. *International Journal of Automation and Computing*, vol. 18, no. 1, pp. 18–44, 2021. DOI: 10.1007/s11633-020-1257-9.

[6] W. Jia, W. Xia, Y. Zhao, H. Min, Y. X. Chen. 2D and 3D palmprint and palm vein recognition based on neural ar-

chitecture search. *International Journal of Automation and Computing*, vol. 18, no. 3, pp. 377–409, 2021. DOI: 10.1007/s11633-021-1292-1.

[7] G. P. Ji, G. B. Xiao, Y. C. Chou, D. P. Fan, K. Zhao, G. Chen, L. Van Gool. Video polyp segmentation: A deep learning perspective. *Machine Intelligence Research*, vol. 19, no. 6, pp. 531–549, 2022. DOI: 10.1007/s11633-022-1371-y.

[8] D. Wu, M. W. Liao, W. T. Zhang, X. G. Wang, X. Bai, W. Q. Cheng, W. Y. Liu. YOLOP: You only look once for panoptic driving perception. *Machine Intelligence Research*, vol. 19, no. 6, pp. 550–562, 2022. DOI: 10.1007/s11633-022-1339-y.

[9] Q. Y. Zhou, C. D. Du, H. G. He. Exploring the brain-like properties of deep neural networks: A neural encoding perspective. *Machine Intelligence Research*, vol. 19, no. 5, pp. 439–455, 2022. DOI: 10.1007/s11633-022-1348-x.

[10] Z. Y. Yang, L. Leng, T. F. Wu, M. Li, J. Chu. Multi-order texture features for palmprint recognition. *Artificial Intelligence Review*, vol. 56, no. 2, pp. 995–1011, 2023. DOI: 10.1007/s10462-022-10194-5.

[11] L. Leng, M. Li, C. Kim, X. Bi. Dual-source discrimination power analysis for multi-instance contactless palmprint recognition. *Multimedia Tools and Applications*, vol. 76, no. 1, pp. 333–354, 2017. DOI: 10.1007/s11042-015-3058-7.

[12] A. W. K. Kong, D. Zhang. Competitive coding scheme for palmprint verification. In *Proceedings of the 17th International Conference on Pattern Recognition*, Cambridge, UK, pp. 520–523, 2004. DOI: 10.1109/ICPR.2004.1334184.

[13] Z. H. Guo, D. Zhang, L. Zhang, W. M. Zuo. Palmprint verification using binary orientation co-occurrence vector. *Pattern Recognition Letters*, vol. 30, no. 13, pp. 1219–1227, 2009. DOI: 10.1016/j.patrec.2009.05.010.

[14] L. K. Fei, Y. Xu, W. L. Tang, D. Zhang. Double-orientation code and nonlinear matching scheme for palmprint recognition. *Pattern Recognition*, vol. 49, pp. 89–101, 2016. DOI: 10.1016/j.patcog.2015.08.001.

[15] Y. Xu, L. K. Fei, J. Wen, D. Zhang. Discriminative and robust competitive code for palmprint recognition. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 2, pp. 232–241, 2018. DOI: 10.1109/TSMC.2016.2597291.

[16] W. Jia, R. X. Hu, Y. K. Lei, Y. Zhao, J. Gui. Histogram of oriented lines for palmprint recognition. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 3, pp. 385–395, 2014. DOI: 10.1109/TSMC.2013.2258010.

[17] Y. T. Luo, L. Y. Zhao, B. Zhang, W. Jia, F. Xue, J. T. Lu, Y. H. Zhu, B. Q. Xu. Local line directional pattern for palmprint recognition. *Pattern Recognition*, vol. 50, pp. 26–44, 2016. DOI: 10.1016/j.patcog.2015.08.025.

[18] L. Zhang, L. D. Li, A. Q. Yang, Y. Shen, M. Yang. Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach. *Pattern Recognition*, vol. 69, pp. 199–212, 2017. DOI: 10.1016/j.patcog.2017.04.016.

[19] W. Jia, B. Zhang, J. T. Lu, Y. H. Zhu, Y. Zhao, W. M. Zuo, H. B. Ling. Palmprint recognition based on complete direction representation. *IEEE Transactions on Image Processing*, vol. 26, no. 9, pp. 4483–4498, 2017. DOI: 10.1109/TIP.2017.2705424.

[20] R. M. Bolle, S. Pankanti, J. H. Connell, N. K. Ratha. Iris individuality: A partial iris model. In *Proceedings of the 17th International Conference on Pattern Recognition*, Cambridge, UK, pp. 927–930, 2004. DOI: 10.1109/ICPR.2004.1334411.

[21] K. P. Hollingsworth, K. W. Bowyer, P. J. Flynn. The best bits in an Iris code. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 6, pp. 964–973, 2009. DOI: 10.1109/TPAMI.2008.185.

[22] L. Zhang, H. Y. Li, J. Y. Niu. Fragile bits in palmprint recognition. *IEEE Signal Processing Letters*, vol. 19, no. 10, pp. 663–666, 2012. DOI: 10.1109/LSP.2012.2211589.

[23] W. K. Kong, D. Zhang, W. X. Li. Palmprint feature extraction using 2-D Gabor filters. *Pattern Recognition*, vol. 36, no. 10, pp. 2339–2347, 2003. DOI: 10.1016/S0031-3203(03)00121-3.

[24] D. Zhang, W. K. Kong, J. You, M. Wong. Online palmprint identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1041–1050, 2003. DOI: 10.1109/TPAMI.2003.1227981.

[25] X. Q. Wu, K. Q. Wang, D. Zhang. Palmprint authentication based on orientation code matching. In *Proceedings of the 5th International Conference on Audio-and Video-Based Biometric Person Authentication*, Hilton Rye Town, USA, pp. 555–562, 2005. DOI: 10.1007/11527923_57.

[26] Z. N. Sun, T. N. Tan, Y. H. Wang, S. Z. Li. Ordinal palmprint represention for personal identification. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, USA, pp. 279–284, 2005. DOI: 10.1109/CVPR.2005.267.

[27] A. Kong, D. Zhang, M. Kamel. Palmprint identification using feature-level fusion. *Pattern Recognition*, vol. 39, no. 3, pp. 478–487, 2006. DOI: 10.1016/j.patcog.2005.08.014.

[28] W. M. Zuo, F. Yue, K. Q. Wang, D. Zhang. Multiscale competitive code for efficient palmprint recognition. In *Proceedings of the 19th International Conference on Pattern Recognition*, Tampa, USA, 2008. DOI: 10.1109/ICPR.2008.4761868.

[29] W. Jia, D. S. Huang, D. Zhang. Palmprint verification based on robust line orientation code. *Pattern Recognition*, vol. 41, no. 5, pp. 1504–1513, 2008. DOI: 10.1016/j.patcog.2007.10.011.

[30] F. Yue, W. M. Zuo, D. Zhang, K. Q. Wang. Orientation selection using modified FCM for competitive code-based palmprint recognition. *Pattern Recognition*, vol. 42, no. 11, pp. 2841–2849, 2009. DOI: 10.1016/j.patcog.2009.03.015.

[31] W. M. Zuo, Z. C. Lin, Z. H. Guo, D. Zhang. The multiscale competitive code via sparse representation for palmprint verification. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Francisco, USA, pp. 2265–2272, 2010. DOI: 10.1109/CVPR.2010.5539909.

[32] Z. Khan, A. Mian, Y. Q. Hu. Contour code: Robust and efficient multispectral palmprint encoding for human recognition. In *Proceedings of International Conference on Computer Vision*, Barcelona, Spain, pp. 1935–1942, 2011. DOI: 10.1109/ICCV.2011.6126463.

[33] Z. N. Sun, L. B. Wang, T. N. Tan. Ordinal feature selection for iris and palmprint recognition. *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3922–3934, 2014. DOI: 10.1109/TIP.2014.2332396.

[34] L. K. Fei, Y. Xu, D. Zhang. Half-orientation extraction of palmprint features. *Pattern Recognition Letters*, vol. 69,

pp. 35–41, 2016. DOI: 10.1016/j.patrec.2015.10.003.

[35] L. K. Fei, B. Zhang, Y. Xu, L. P. Yan. Palmprint recognition using neighboring direction indicator. *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 6, pp. 787–798, 2016. DOI: 10.1109/THMS.2016.2586474.

[36] M. Tabejamaat, A. Mousavi. Concavity-orientation coding for palmprint recognition. *Multimedia Tools and Applications*, vol. 76, no. 7, pp. 9387–9403, 2017. DOI: 10.1007/s11042-016-3544-6.

[37] Q. Zheng, A. Kumar, G. Pan. Suspecting less and doing better: New insights on palmprint identification for faster and more accurate matching. *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 3, pp. 633–641, 2016. DOI: 10.1109/TIFS.2015.2503265.

[38] F. Ma, X. K. Zhu, C. L. Wang, H. J. Liu, X. Y. Jing. Multi-orientation and multi-scale features discriminant learning for palmprint recognition. *Neurocomputing*, vol. 348, pp. 169–178, 2019. DOI: 10.1016/j.neucom.2018.06.086.

[39] L. Leng, Z. Y. Yang, W. D. Min. Democratic voting downsampling for coding-based palmprint recognition. *IET Biometrics*, vol. 9, no. 6, pp. 290–296, 2020. DOI: 10.1049/iet-bmt.2020.0106.

[40] Z. Y. Yang, L. Leng, W. D. Min. Extreme downsampling and joint feature for coding-based palmprint recognition. *IEEE Transactions on Instrumentation and Measurement*, vol. 70, Article number 5005112, 2021. DOI: 10.1109/TIM.2020.3038229.

[41] G. Li, J. Kim. Palmprint recognition with Local Microstructure Tetra Pattern. *Pattern Recognition*, vol. 61, pp. 29–46, 2017. DOI: 10.1016/j.patcog.2016.06.025.

[42] L. K. Fei, B. Zhang, W. Zhang, S. H. Teng. Local apparent and latent direction extraction for palmprint recognition. *Information Sciences*, vol. 473, pp. 59–72, 2019. DOI: 10.1016/j.ins.2018.09.032.

[43] L. K. Fei, B. Zhang, Y. Xu, Z. H. Guo, J. Wen, W. Jia. Learning discriminant direction binary palmprint descriptor. *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 3808–3820, 2019. DOI: 10.1109/TIP.2019.2903307.

[44] L. K. Fei, B. Zhang, Y. Xu, D. Huang, W. Jia, J. Wen. Local discriminant direction binary pattern for palmprint representation and recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 2, pp. 468–481, 2020. DOI: 10.1109/TCSVT.2019.2890835.

[45] L. Zhang, M. Yang, X. C. Feng. Sparse representation or collaborative representation: Which helps face recognition? In *Proceedings of International Conference on Computer Vision*, Barcelona, Spain, pp. 471–478, 2011. DOI: 10.1109/ICCV.2011.6126277.

[46] K. P. Hollingsworth, K. W. Bowyer, P. J. Flynn. Improved iris recognition through fusion of hamming distance and fragile bit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2465–2476, 2011. DOI: 10.1109/TPAMI.2011.89.

[47] D. Zhang, Z. H. Guo, G. M. Lu, L. Zhang, W. M. Zuo. An online system of multispectral palmprint verification. *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 2, pp. 480–490, 2010. DOI: 10.1109/TIM.2009.2028772.

[48] W. Jia, R. X. Hu, J. Gui, Y. Zhao, X. M. Ren. Palmprint recognition across different devices. *Sensors*, vol. 12, no. 6, pp. 7938–7964, 2012. DOI: 10.3390/s120607938.

[49] A. Kumar, S. Shekhar. Personal identification using multibiometrics rank-level fusion. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 5, pp. 743–752, 2011. DOI: 10.1109/TSMCC.2010.2089516.

Haitao Wang received the B. Sc. degree in software engineering (mechanical and electronic engineering) from Jiangxi University of Science and Technology, China in 2020. He is a master student in School of Computer Science and Information Engineering, Hefei University of Technology, China.

His research interests include palmprint recognition, computer vision and deep learning.

E-mail: 2020171079@mail.hfut.edu.cn

ORCID iD: 0009-0005-2513-0746

Wei Jia received the B. Sc. degree in informatics from Central China Normal University, China in 1998, the M. Sc. degree in computer science from Hefei University of Technology, China in 2004, and the Ph. D. degree in pattern recognition and intelligence systems from University of Science and Technology of China, China in 2008. He has been a research associate professor in Hefei Institutes of Physical Science, Chinese Academy of Sciences, China from 2008 to 2016. He is currently a full professor in the School of Computer Science and Information Engineering, Hefei University of Technology, China.

His research interests include computer vision, biometrics, pattern recognition, image processing and machine learning.

E-mail: jiawei@hfut.edu.cn (Corresponding author)

ORCID iD: 0000-0001-5628-6237

Machine Intelligence Research

Citation: H. Wang, W. Jia. Enhance the performance of directional feature–based palmprint recognition by directional response stability measurement. *Machine Inteligence Research*, vol.21, no.3, pp.597–614, 2024. https://doi.org/10.1007/s11633–023–1436–6