Three Dimensional Palmprint Recognition with Joint Line and Orientation Features

Wei Li, David Zhang, Lei Zhang, Guangming Lu, and Jingqi Yan

ABSTRACT Two dimensional (2D) palmprint has been recognized as an effective biometric identifier in the past decade. Recently, three dimensional (3D) palmprint recognition was proposed to further improve the performance of palmprint systems. This paper presents a simple yet efficient scheme for 3D palmprint recognition. After calculating and enhancing the Mean Curvature Image (MCI) of the 3D palmprint data, we extract both line and orientation features from it. The two types of features are then fused at either score level or feature level for the final 3D palmprint recognition. The experiments on the HKPU 3D palmprint database which contains 8000 samples from 400 palms show that the proposed feature extraction and fusion methods lead to promising performance.

KEYWORDS 3D palmprint identification; biometrics; feature fusion; Mean curvature

1 Introduction

Automatic personal authentication using biometric characteristics plays a key role in applications of public security, access control, forensics and e-banking, etc. Many kinds of biometric authentication techniques have been developed based on different biometric characteristics, such as fingerprint, face, iris, palmprint, hand shape, etc. Two dimensional (2D) palmprint recognition has been widely studied in the past decade and it has been proven that palmprint is a unique biometric identifier. 2D palmprint systems have merits of high accuracy and user friendliness, etc. Nonetheless, 2D palmprint can be easily counterfeited and much three dimensional (3D) palm structural information is lost. Inspired by the success of 3D techniques in biometric authentication, such as 3D face and 3D ear recognition, very recently a structured-light imaging based 3D palmprint system was developed to capture the depth information of palmprint. In [9], the Mean curvature and Gaussian curvature are calculated from the depth information and they serve as the basic features for 3D palmprint matching and recognition.

As shown in [9], the Mean curvature is a stable and distinct feature of 3D palmprint. By normalizing and mapping the Mean curvature values to a plane, we can get a Mean Curvature Image (MCI) which contains line structure features and texture features of the 3D palmprint. In [9], the MCI was binarized to highlight the line features and the binarized MCI was used as the feature map for 3D palmprint matching. However, the binarization operation loses much the texture information existing in the MCI. Actually, if we view the MCI of a 3D palmprint as a 2D palmprint image, then many 2D palmprint feature extraction techniques can be applied. In addition, the line and texture features could provide complementary information for palmprint discrimination. Therefore, in this paper we propose to extract both line and texture features from MCI and fuse them efficiently for more accurate 3D palmprint recognition.

There are two main approaches to 2D palmprint recognition: line-based approach and texture-based approach. For the representative line-based methods, Li et al. proposed a modified line-based Hausdorff Distance for palmprint identification; Wu et al. proposed a set of directional line detectors and used them to extract the palm lines for palmprint matching;...
Huang et al. proposed a modified finite Radon transform to extract the principal lines in palmprint [19]. With respect to the texture-based methods, the most representative one may be Competitive Coding (CompCode) scheme proposed by Kong et al. [2], where a series of directional Gabor filters were used to extract the orientation features of palmprint. Sun et al. proposed an ordinal palmprint representation for personal identification [3].

With the MCI of 3D palmprint data, the line features can be easily extracted by setting a global threshold to segment the high curvature regions. For texture features, we can use six directional Gabor filters to extract the local orientation from the MCI, like what the CompCode method does on 2D palmprint images [2]. In our preliminary work [19], the line and orientation features are directly extracted from the MCI and they are fused at matching score level. In this paper, we propose to use the Butterworth Low-Pass Filter (BLPF) to enhance the MCI before feature extraction, and fuse the extracted line and orientation features on the feature level. A series of experiments are conducted by using the HKPU 3D palmprint database, which contains 8000 samples collected from 400 palms. The experimental results show that the proposed method is very promising, outperforming significantly the methods in [9] and [19]. Particularly, the proposed feature-level fusion can not only achieve higher accuracy than the score level fusion presented in [19], but also require much less matching time.

The rest of the paper is organized as follows. Section 2 discusses the calculation of MCI. Section 3 introduces the line and orientation feature extraction. Section 4 presents the matching and fusion scheme. Section 5 presents the experimental results and Section 6 concludes the paper.

2 Mean Curvature Image Processing

2.1 Region of Interest (ROI) Extraction

In [9], we have developed a structured-light imaging based 3D palmprint acquisition device. The 3D palmprint data are typical range data which are represented by cloud points. Fig. 1c shows a 3D palmprint sample (resolution: 768x576) collected by our device. For a better visualization of the 3D palmprint, Fig. 1c is rendered by OpenGL. We can see that in the 3D palmprint image, the cloud points in the boundary area and in the fingers are not suitable for feature extraction and recognition. Most of the useful and stable features locate in the center area of the palm. In addition, at different times when the user puts his/her hand on the collecting device, there will be some relative displacements of the positions of palm, even that we impose some constraints on the users to place their hands. Therefore, before feature extraction it is necessary to perform some preprocessing to align the palmprint and extract the central area of it, which is called the Region of Interest (ROI) extraction.

Figure 1. The ROI extraction of 3D palmprint from its 2D counterpart. (a) The 2D palmprint image, the adaptively established coordinate system and the ROI (i.e. the rectangle); (b) the extracted 2D ROI; (c) the 3D palmprint image, whose cloud points have a one-to-one correspondence to the pixels in the 2D counterpart; (d) the obtained 3D ROI by extracting the cloud points corresponding to the pixels in 2D ROI.

Our 3D palmprint data acquisition device can capture a 3D palmprint image and a 2D palmprint image simultaneously. As in [9], we extract the 3D ROI with the aid of its corresponding 2D counterpart. Fig. 1a shows a 2D palmprint image, the established local coordinate system by using the algorithm in [1] and the ROI (i.e. the rectangle). Fig. 1b shows the extracted 256x256 2D ROI. Because the points in 3D palmprint image have a one to one correspondence to the points in its 2D counterpart, we can easily obtain the 3D ROI by extracting the cloud points corresponding to the 2D ROI. Fig. 1d shows the extracted 3D ROI from Fig. 1c. (Note that we use a different viewpoint to show the 3D ROI in Fig. 1d.) By using the ROI extraction procedure, the 3D palmprint images are aligned so that the small translation and rotation introduced in the data acquisition process are corrected. In addition, the data amount used in the following feature extraction and matching process is significantly reduced. This will save much computational cost.

2.2 Curvature Calculation

With the ROI obtained from the original 3D palmprint data, stable and unique features are expected to be extracted for the following pattern matching and recognition. The Mean and Gaussian curvatures are intrinsic measures of a surface, i.e. they depend only on the surface shape but not on the way how the surface is placed in the 3D space [10]. Thus such curvature features are robust to the rotation, translation and even some deformation of the palm. From the work in [9], we know that the Mean curvature is more informative than the Gaussian curvature in 3D palmprint recognition. Thus to save computation, we only consider the Mean curvature in the following development. We adopt the algorithm in
[11] to estimate the Mean curvature from 3D palmprint data for its simplicity and effectiveness:

\[ H = \frac{(1 + (h_x)^2)h_{yy} - 2h_xh_yh_{xx} + (1 + (h_y)^2)h_{xx}}{2(1 + (h_x)^2 + (h_y)^2)^{3/2}} \]  

(1)

where \( h \) is the height of the points on the palmprint to the reference plane, \( h_x, h_y, h_{xx}, h_{yy} \) and \( h_{xy} \) are the first, second and hybrid partial derivatives of \( h \) to \( x \) and \( y \) coordinates separately.

With Eq. (1), the Mean curvatures of a 3D palmprint ROI can be calculated. For better visualization and more efficient computation, we convert the original curvature images into grey level images with integer pixels. We first normalize the Mean curvature value \( H \) to 0.5 as follows

\[ \tilde{H}(i,j) = 0.5(H(i,j) - \mu) + 0.5 \]

(2)

where \( \mu \) and \( \delta \) are the mean and standard deviation of the curvature value. With Eq. (2), most of the curvature values will be normalized into the interval [0,1]. We then map \( \tilde{H}(i,j) \) to an 8-bits grey level image \( G(i,j) \):

\[
G(i,j) = \begin{cases} 
0 & \tilde{H}(i,j) \leq 0 \\
\text{round}(255 \times \tilde{H}(i,j)) & 0 < \tilde{H}(i,j) < 1 \\
255 & \tilde{H}(i,j) \geq 1
\end{cases}
\]

(3)

We call image \( G(i,j) \) the Mean Curvature Image (MCI). Fig. 2 illustrates the MCI images from the same palm (at different times) and different palms. We can see that the 2D MCI images can well preserve the 3D palm surface features. Not only the principal lines, which are the most important and stable features in palmprint recognition, are clearly enhanced in MCI, but also the depth information of different shape structures is well preserved.

3.2 Noise Removal

In the 3D palmprint data acquisition process, noise will be inevitably introduced. This can be clearly observed in the MCI images in Fig. 2. The noise mainly comes from two sources. One is the system electrical circuit hardware. Such system noise is often the mixture of high frequency periodical noise and white noise. The other source is the imaging object, i.e. the palm. The palm is not a rigid object and its small deformations in the data collection process add random noise to the collected data. It is necessary to remove noise from the raw MCI images for a robust feature extraction. Considering the fact that these noise will mainly fall into the high frequency band, we simply use a Butterworth Low-Pass Filter (BLPF) to reduce them. The BLPF is defined as:

\[ H(u,v) = \frac{1}{1 + [D(u,v)/D_0]^n}, \quad u = 1, 2, \cdots, M, \quad v = 1, 2, \cdots, N \]

(4)

where \( M \times N \) is the image size; \( D_0 \) is the cut-off frequency (20 is used in our experiments); \( n \) is the order of BLPF and we set it to 4 by experience; \( D[u,v] \) is defined as:

\[ D(u,v) = [(u - M/2)^2 + (v - N/2)^2]^{1/2} \]

(5)

Denote by \( G \) the noisy MCI, the denoised MCI is obtained as follows:

\[ G' = \text{IFT}(\text{FT}(G) \cdot H) \]

(6)

where \( \text{FT} \) denotes Fourier Transform, and \( \text{IFT} \) denotes Inverse Fourier Transform. Fig. 3 compares the MCI images before and after noise removal, and we can clearly see that the image quality is much improved, which will benefit greatly the following feature extraction and matching.

Figure 3. Noise removal of MCI. The first row is the original MCI, and the second row is the MCI filtered by BLPF.

Figure 4. The binarized MCI images. The white areas represent the high Mean curvature region position.

3.1 Line Feature Extraction

The principal lines and strong wrinkles are the most stable and significant features in the palmprint.
3D palmprint, these features are represented by high curvature regions. So, it’s very easy to extract the line feature from MCI by thresholding:

\[
L(i,j) = \begin{cases} 
1 & G'(i,j) < c \cdot \mu_G \\
0 & \text{otherwise}
\end{cases}
\]

where \( c \) is a constant and \( \mu_G \) is the mean value of \( G'(i,j) \). We set \( c = 0.7 \) in the experiments by experience. Note that binary image \( L \) can be directly used for matching. Fig. 4 shows the binarized images of the MCI images in Fig. 3.

### 3.2 Orientation Feature Extraction

The line features extracted in Section 3.1 can indicate where the significant structures will happen in a palm, but the orientations of these line features are not implicitly represented. Apart from the line features, the MCI also has many finer texture features, which can be well characterized by local orientations as what has been done in 2D palmprint recognition \(^3\). The Gabor filters have excellent capability to extract such features. By convolving the MCI with a series of Gabor filters along different orientations, the orientation along which the Gabor filter has the greatest response can be taken as the orientation of that point. The orientation features can then be coded and matched by angular distance for identification. This process is called the Competitive Coding scheme \(^2\). In this paper, the following Gabor filter is used for extracting the orientations \(^{11}\):

\[
\psi(x, y, \omega, \theta) = \frac{\omega}{\sqrt{2\pi\kappa}} e^{-\frac{(x' + y')^2}{2\kappa^2}} (e^{i\omega x'} - e^{-\frac{x'^2}{2}})
\]

where \( x' = (x - x_0)\cos \theta + (y - y_0)\sin \theta \), \( y' = -(x - x_0)\sin \theta + (y - y_0)\cos \theta \), and \( (x_0, y_0) \) is the center of the function; \( \theta \) is the orientation of the Gabor functions in radians; \( \omega = \kappa / \sigma \) is the radial frequency in radians per unit length. We set \( \sigma = 4.2 \) by experience, while \( \kappa \) is a coefficient defined by

\[
\kappa = \sqrt{2\ln 2 \left( \frac{2^\alpha + 1}{2^\alpha - 1} \right)}
\]

where \( \alpha \) is the half-amplitude bandwidth of the frequency response. Here, we choose \( \alpha = 1.3785 \) octave by experience. More information about Gabor filters can be found in \(^{13}\). In this paper we set the size of Gabor filter template as \( 35 \times 35 \) with the center position \((17, 17)\).

Based on our experiments (please refer to section 5 for details), we choose to use six Gabor filters with orientations \( \theta = 0, \pi / 6, 2\pi / 6, 3\pi / 6, 4\pi / 6, 5\pi / 6 \) in the implementation for a good balance of accuracy and efficiency. Convolving the six filters with the MCI, and selecting the orientation which leads to the greatest response, we get the orientation features of MCI. Fig. 5 shows an example, from which we can see that the extracted orientations can well represent the local directional structure in a neighborhood.

### 4 Line and Orientation Feature Matching and Fusion

#### 4.1 Feature Matching

For the binary line feature map, we use the AND operation to calculate the matching score between two maps. Denote by \( L_d \) the binary MCI image in the database and by \( L_t \) the input MCI image. Suppose the image size is \( n \times m \). The matching score between \( L_d \) and \( L_t \) is defined as:

\[
R_{\land} = \frac{2\sum_{i=1}^{n} \sum_{j=1}^{m} L_d(i,j) \land L_t(i,j)}{\sum_{i=1}^{n} \sum_{j=1}^{m} L_d(i,j) + \sum_{i=1}^{n} \sum_{j=1}^{m} L_t(i,j)}
\]

where symbol “\( \land \)” means the AND logic operation. If \( L_d \) and \( L_t \) are identical, we will have the maximum matching score \( R_{\land} = 1 \); on the contrary, if \( L_d \) and \( L_t \) are extremely different, the matching score will be \( R_{\land} = 0 \).

For orientation features, we use integers 0~5 to code the six orientations \( 0, \pi / 6, 2\pi / 6, 3\pi / 6, 4\pi / 6, 5\pi / 6 \), respectively. Intuitively, the distance between parallel orientations should be 0, while the distance between perpendicular orientations should be 3. In other cases, the distance should be 1 or 2. Let \( D_d \) and \( D_t \) be the direction sets of the MCI images. The matching score between them can be defined as:

\[
R_{\oplus} = \frac{1}{3nm} \sum_{i=1}^{n} \sum_{j=1}^{m} F(D_d(i,j), D_t(i,j))
\]

where \( F(\alpha, \beta) \) represents the angular distance between

![Figure 5. The orientation map of an MCI.](image)
\( F(\alpha, \beta) = \min(\alpha - \beta, 6 - |\alpha - \beta|), \ \alpha, \beta \in \{0,1,2,3,4,5\} \) \tag{12}

Obviously, the value of \( F(\alpha, \beta) \) can only be 0, 1, 2 or 3 as described above.

### 4.2 Fusion Scheme

#### 4.2.1 Score level fusion

Suppose there are \( n \) matching scores and denote them by \( R_i, \ i = 1, 2, \ldots, n \). The commonly used score level fusion techniques include Min-Score (MIN) \( R_{\text{MIN}} = \min(R_1, R_2, \ldots, R_n) \), Max-Score (MAX) \( R_{\text{MAX}} = \max(R_1, R_2, \ldots, R_n) \), Summation (SUM) \( R_{\text{SUM}} = \frac{1}{n} \sum_{i=1}^{n} R_i \), and Weighted Average (WA) methods \([14-15]\). Because the EER (Equal Error Rate) is an important index of the matching result and it can be estimated by the training database, the weights can be determined according to the corresponding EER values. In [14], a WA scheme, called Matcher Weighting (MW), is proposed:

\[
R_{\text{MW}} = \frac{1}{n} \sum_{i=1}^{n} w_i R_i \quad w_i = \frac{1}{\sum_{j=1}^{n} e_j} \tag{13}
\]

where \( w_i \) is the weight of \( R_i \) and \( e_i \) is the corresponding EER. The MW scheme assigns smaller weights to those features with higher EER values. Here, we adopt this fusion scheme to fuse the matching scores obtained by line and orientation features.

#### 4.2.2 Feature Level Fusion

For line and orientation features, after coding them to bit planes, it’s very convenient to combine these bit planes for matching. The six types of orientation features can be coded to 3-bits planes as illustrated in Table 1 [5]. The binary line feature map can be readily represented by another bit plane (i.e. the 4th bit). Then, for each point in the MCI, we can use a 4-bits code to describe the line and orientation features. Let \( B_j \) and \( B_t \) be the 4-bits feature maps of two MCIs. The matching score between them can be efficiently calculated as follows:

\[
R_j = \sum_{k} \sum_{i} \left( (B_j^k(i,j) \cap B_t^k(i,j)) \cap (B_j^k(i,j) \oplus B_t^k(i,j)) \right) \tag{14}
\]

where \( B_j^k \) and \( B_t^k \) are two masks which indicate the non-palmprint pixels, \( B_j^k \) and \( B_t^k \) represent the \( k \)-th bit plane of \( B_j \) and \( B_t \), and \( \oplus \) is the bitwise exclusive OR operation.

### 5 Experimental Results

In [9], a 3D palmprint database has been established by using the 3D palmprint imaging device developed by the Biometrics Research Center, The Hong Kong Polytechnic University. The database is available at http://www4.comp.polyu.edu.hk/~biometrics/2D_3D_Palmprint.htm. The PolyU 3D palmprint database contains 8000 samples from 200 volunteers, including 136 males and 64 females between 10 and 55 years old. The 3D palmprint samples were collected in two separated sessions, and in each session 10 samples were collected from both the left and right hands of each subject. The average time interval between the two sessions is one month. The original spatial resolution of the data is 768x576. After ROI extraction, the central part (256x256) is used for feature extraction and recognition. The z-value resolution of the data is 32 bits. In data collection, each volunteer contributed samples from both his/her right-hand palm and left-hand palm. Samples collected from the same palm belong to the same class. Therefore, there are 400 classes and each class contains 20 samples in our database.

We performed two types of experiments on the established database: verification and identification. The experiments were performed under the Visual C++ 6.0 programming environment on a PC with Windows XP Professional operation system and Pentium 4 CPU of 2.66GHz and 1GB RAM. In verification, the class of the input palmprint is known and each of the 3D samples was matched with all the other 3D samples in the database. A successful matching is called intra-class matching or genuine if the two samples are from the same class. Otherwise, the unsuccessful matching is called inter-class matching or impostor. Using the established database, there are 31,996,000 matchings in total.

In extracting the orientation features, the number of Gabor filters (refer to section 3.2) should be determined. To this end, we performed a series of experiments by using different numbers of Gabor filters to extract the orientation features on the 3D palmprint database. The orientation \( \theta \) of the Gabor filters is evenly partitioned in \([0, \pi]\). For example, if four Gabor filters are used, we have \( \theta = 0, \pi/4, 2\pi/4, 3\pi/4 \). Table 2 lists the equal error rate (EER) results by using 4–12 Gabor filters. We can...
see that the lowest EER is got by six Gabor filters. Using more than six Gabor filter cannot improve the accuracy but increase the computational cost. Therefore we use six Gabor filters (θ = 0, π/6, 2π/6, 3π/6, 4π/6, 5π/6) in the following experiments.

The verification experiments were performed by using each of the line and orientation features, as well as the fusion of them at score level and feature level respectively. We compared the proposed methods with the MW method in [9], which fuses MCI, GCI and ST features on score level, and the score level fusion method in [19]. The ROC curves are shown in Fig. 6. The EER values are listed in Table 3, where the feature size, the preprocessing, feature extraction and matching time by using different features are also listed. The preprocessing in this paper includes ROI extraction and BLPF based noise removal, while the preprocessing in [9] and [19] only includes ROI extraction. The BLPF filtering costs about 0.5s but it improves the recognition accuracy significantly. From Fig. 6 and Table 3, we can see that the proposed fusion methods get much lower EER than other methods. Between score level fusion and feature level fusion, the later can achieve slightly better EER while requiring only half of the matching time. This implies that fusing the line and orientation features at feature level could be a more practical solution to real time 3D palmprint identification in a relatively large scale database.

The experiments of identification were also conducted on the 3D palmprint database. In identification, we do not know the class of the input palmprint but want to identify which class it belongs to. In the experiments we let the first sample of each class in the database be template and use the other samples as probes. Therefore, there are 7600 probes and 400 templates. The probes were matched with all the templates models, and for each probe, the matching results were ordered according to the matching scores. Then we can get the cumulative match curves as shown in Fig. 7. The cumulative matching performance, rank-one recognition rate and lowest rank of perfect recognition (i.e. the lowest rank

Table 2. EER by using different numbers of Gabor filters to extract the orientation features on 3D palmprint database.

<table>
<thead>
<tr>
<th>Number of Gabor filters</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>EER (%)</td>
<td>0.56</td>
<td>0.38</td>
<td>0.32</td>
<td>0.33</td>
<td>0.39</td>
<td>0.35</td>
<td>0.34</td>
<td>0.36</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table 3. Verification performance and running time by different methods.

<table>
<thead>
<tr>
<th></th>
<th>Line feature</th>
<th>Orientation feature</th>
<th>Score level fusion</th>
<th>Feature level fusion</th>
<th>MW[9]</th>
</tr>
</thead>
<tbody>
<tr>
<td>With BLPF</td>
<td>Yes</td>
<td>No [19]</td>
<td>Yes</td>
<td>No [19]</td>
<td>Yes</td>
</tr>
<tr>
<td>Feature size (byte)</td>
<td>1568</td>
<td>1568</td>
<td>512</td>
<td>512</td>
<td>2080</td>
</tr>
<tr>
<td>EER</td>
<td>0.53%</td>
<td>0.688%</td>
<td>0.32%</td>
<td>0.495%</td>
<td>0.17%</td>
</tr>
<tr>
<td>Preprocessing time</td>
<td>1.116s</td>
<td>660ms</td>
<td>1.116s</td>
<td>660ms</td>
<td>1.116s</td>
</tr>
<tr>
<td>Feature extraction time</td>
<td>96ms</td>
<td>96ms</td>
<td>97ms</td>
<td>97ms</td>
<td>183ms</td>
</tr>
<tr>
<td>Matching time</td>
<td>0.35 ms</td>
<td>0.35 ms</td>
<td>0.15ms</td>
<td>0.15ms</td>
<td>0.50ms</td>
</tr>
</tbody>
</table>

Table 4. Identification performance by different methods.

<table>
<thead>
<tr>
<th></th>
<th>Line feature</th>
<th>Orientation feature</th>
<th>Score level fusion</th>
<th>Feature level fusion</th>
<th>MW[9]</th>
</tr>
</thead>
<tbody>
<tr>
<td>With BLPF</td>
<td>Yes</td>
<td>No [19]</td>
<td>Yes</td>
<td>No [19]</td>
<td>Yes</td>
</tr>
<tr>
<td>Rank-one recognition rate</td>
<td>98.71%</td>
<td>98.46%</td>
<td>99.24%</td>
<td>99.11%</td>
<td>99.76%</td>
</tr>
<tr>
<td>Lowest rank for perfect recognition</td>
<td>57</td>
<td>71</td>
<td>39</td>
<td>46</td>
<td>28</td>
</tr>
</tbody>
</table>

Figure 6. ROC curves by different methods.

Figure 7. CMC curves by different methods.
when the recognition rate reaches 100%) are listed in Table 4. From the experimental results we can see that the performance of the proposed fusion scheme is much better than the other three methods.

6 Conclusions

This paper presented simple yet efficient schemes to extract and fuse the line and orientation features of 3D palmprint for recognition. After the 3D palmprint image was captured, the region of interest (ROI) was extracted, from which the Mean Curvature Image (MCI) was calculated. By using the Butterworth Low-Pass Filter (BLPF) to remove the high frequency noise, we extracted the line and orientation features from MCI, which are robust features for palmprint recognition. The score level and feature level fusion of the two types of features were proposed to match and classify the palmprints. A series of verification and identification experiments were performed on the HKPU 3D palmprint database with 8000 samples from 200 individuals (400 palms). The experimental results showed that both the score level and feature level fusion of line and orientation features can have much better results than using only one of them and the existing 3D palmprint recognition methods. Particularly, the feature level fusion of line and orientation features can achieve better accuracy than score level fusion while requiring less matching time.

References


作者简介

黎伟，2010年于上海交通大学模式识别与智能系统专业获工学博士学位，现任深圳先进技术研究院助理研究员。研究方向：生物特征识别、图像处理，虚拟化，海量数据挖掘等。已在国际期刊，如IEEE Transactions，以及国际会议，如CVPR等，发表多篇相关学术论文。