

Received May 7, 2019, accepted May 15, 2019, date of publication May 24, 2019, date of current version June 19, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2918778

Extracting Palmprint ROI From Whole Hand Image Using Straight Line Clusters

QIANWEN XIAO^{1,2}, JINGTING LU³, WEI JIA^{1,2}, (Member, IEEE), AND XIAOPING LIU^{1,2}

¹School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China

²Anhui Province Key Laboratory of Industry Safety and Emergency Technology, Hefei 230011, China

³Institution of Industry and Equipment Technology, Hefei University of Technology, Hefei 230009, China

Corresponding author: Xiaoping Liu (liu@hfut.edu.cn)

This work was supported in part by the National Science Foundation of China 61673157, in part by the Fundamental Research Funds for the Central Universities under Grant JZ2017HGTA0178 and Grant PA2018GDQT0014, and in part by the Key Research and Development Program in Anhui Province under Grant 1804a09020036.

ABSTRACT This paper proposes a novel method to extract a palmprint region of interest (ROI) from whole hand images using straight line clusters. The core idea of our method is that the fingers can be easily detected by straight line clusters due to the distinctive appearance of the fingers. In our method, the original whole hand image is first converted to a binary image. In the binary image, we draw many straight lines according to several predefined rules. For one straight line, if there are eight intersection points between this straight line and the hand region, we can conclude that this line passes through four fingers. In this case, it is easy to know the positions of finger joint areas. Then, key point candidates can be further detected in these finger joint areas. In our method, we draw many straight lines to detect finger joint areas, which may result in detecting several different key point candidates in one finger joint area. Thus, a lot of key point candidates may be obtained in four finger joint areas. We then exploit the k-means clustering algorithm to calculate four cluster centers, which are treated as the final four key points. Furthermore, utilizing the distance information among four key points, we can know the position order of four key points. The final key points can be used to construct a coordinate system. In this new coordinate system, after rotation normalization, the ROI can be extracted from the central region of hand. We also collected a database including 16 000 whole hand images. The experimental results demonstrate that the proposed method can achieve 100% localization and extraction accuracy.

INDEX TERMS Biometrics, palmprint, region of interest, extraction, straight line clusters.

I. INTRODUCTION

With the rapid development of digital society, personal authentication becomes more and more important for many applications. It is well known that biometrics is one of the most effective solutions for personal authentication [1]–[3]. Biometrics is a field of technology that uses automated methods for identifying or verifying a person based on a physiological or behavioral trait, which has shown significant advantages over traditional personal authentication mechanisms, such as keys, passwords, personal identification numbers, and smart cards [1]–[2]. So far, many human traits have been investigated in depth for the purpose of personal authentication including face, iris, fingerprint, voice

The associate editor coordinating the review of this manuscript and approving it for publication was Jiachen Yang.

and signature, etc. Face recognition, fingerprint recognition and iris recognition are three successful biometrics technologies, which have been widely used for all kinds of real applications. However, they still have some weak points. Thus, researchers try to investigate other effective biometrics technologies. In recent years, as one of emerging biometrics technologies, palmprint recognition has received much attention [4]–[33]. Some previous researches have shown that, compared with fingerprint or iris-based personal biometrics systems, palmprint-based biometrics system has several special advantages, such as stable line features, less distortion, and easy self-positioning [4]–[8]. And, it can also achieve high-recognition rate with fast processing speed. For the aforementioned reasons, nowadays the research related to palmprint recognition is becoming more and more active.

A typical biometric recognition system has two stages of operation, namely, the enrollment stage and the recognition stage. In the enrollment stage, the biometric system acquires the biometric trait of an individual, extracts a salient feature set from it and stores the extracted feature set in a database (often referred to as a template), along with an identifier associating the feature set with an individual. During the recognition stage, the system once again acquires the biometric trait of an individual, extracts a feature set from it, and compares this feature set against the templates in the database in order to determine a match or to verify a claimed identity. Usually, in the recognition stage, there are the following operation steps: (1) data acquisition, (2) preprocessing including Region of Interest (ROI) extraction and quality estimation [34]–[35], (3) feature extraction, (4) matching, and (5) fusion. In those image-based biometrics systems, the location and extraction of ROI is one of key steps. For the palmprint recognition, the purpose of ROI extraction is to adjust the hand images for rotation and scale normalization, and then crop the central region from the normalized hand image for the following feature extraction. Obviously, good ROI extraction algorithm is very crucial for palmprint recognition because the quality of ROI extraction has a very big influence on the recognition performance. Therefore, designing an effective ROI extraction algorithm is an important topic in the research of palmprint recognition.

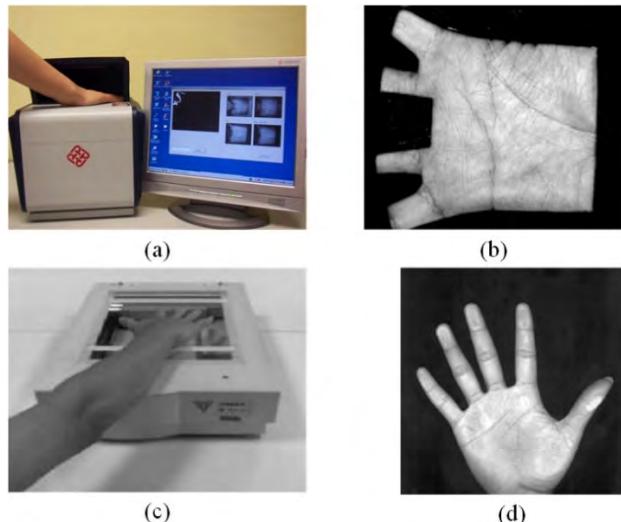


FIGURE 1. Contact hand image acquisition. (a) The contact palmprint acquisition device designed by the Hong Kong Polytechnic University (PolyU) [30]. (b) Part hand image captured by (a) [30]. (c) The scanner used to capture hand image [31]. (d) whole hand image captured by (c) [31].

According to whether the hand touches the capture device or not, the palmprint acquisition modes can be divided into two categories, i.e., contact-based acquisition as shown in Fig. 1 [30], [31] and contactless-based acquisition as shown in Fig. 2 [32], [33]. And, according to whether the captured image contains the whole hand or not, the palmprint acquisition modes can be divided into two categories,

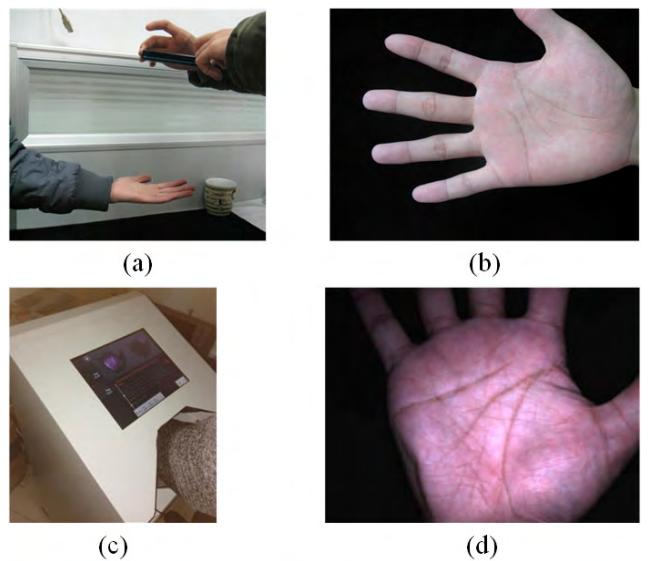


FIGURE 2. Contactless hand image acquisition. (a) Using a smart phone to capture the whole hand image [32], (b) whole hand image captured by (a) [32], (c) the contactless hand image acquisition device designed by the Tongji University [33], (d) part hand image captured by (c) [33].

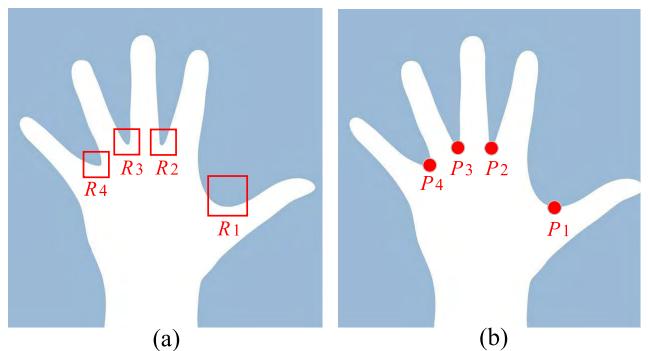


FIGURE 3. Four finger joint areas and corresponding key points. (a) Four finger joint areas R_1 , R_2 , R_3 , and R_4 , (b) four corresponding key points P_1 , P_2 , P_3 , and P_4 .

i.e., whole hand-based acquisition as shown in Fig. 1(d) and Fig. 2(b) and part hand-based acquisition as shown in Fig. 1(b) and Fig. 2(d). So far, the main strategy of most palmprint ROI extraction methods is to extract ROI based on the key points in finger joint areas. As shown in Fig. 3(a), a hand has four finger joint areas, which are R_1 , R_2 , R_3 , and R_4 . The definitions of four finger joint areas are presented as follows: R_1 , the joint area between thumb and the index finger; R_2 , the joint area between the index finger and the middle finger; R_3 , the joint area between the middle finger and the ring finger; R_4 , the joint area between the ring finger and the little finger. In each finger joint area, there is one key point. As shown in Fig. 3(b), these four key points are P_1 , P_2 , P_3 , and P_4 . The mainstream methods of palmprint ROI extraction usually use the key points of P_2 and P_4 to construct coordinate system, and then crop the ROI subimage. Tangent based method [30] and the contour profile distance distribution-based method [31]

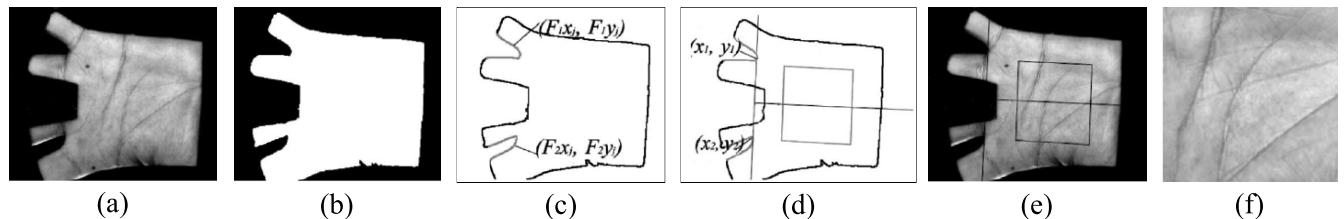


FIGURE 4. The main steps of tangent-based method [30]. (a) Original hand image, (b) binary image, (c) the boundaries of the finger joint areas, (d) compute the tangent of the two gaps and determine two key points, (e) the new coordinate system in hand image, and (f) extracted ROI subimage.

are two main and effective palmprint ROI extraction algorithms, but they still have some limitations. For example, tangent-based method needs to know the approximate positions of R_2 and R_4 in advance [30]. Although the contour profile distance distribution-based method can be used to extract ROI from the whole hand image, it has the follow two drawbacks: (1) in some hand images, the reference point of hand is difficult to be determined; (2) in some hand images with poor image quality, the heavy noise in contour profile often leads to the failure of ROI extraction. In the research of palmprint recognition, capturing whole hand is the trend of development. The main reason is that the whole hand image contains more information of different modalities including palmprint, hand shape and finger knuckle print, which is easy to be used to design a robust multimodal biometrics system. Therefore, to design effective palmprint ROI extraction method from the whole hand image is of significance.

In this paper, we propose a novel method to extract palmprint ROI from the whole hand image. The core idea is using dense straight line clusters to effectively detect finger joint areas. In our method, the original hand image is first converted to a binary image. In binary image, we draw many straight lines according to several predefined rules. For one straight line, if there are 8 or more intersection points between this straight line and hand region, we can conclude that this line passes through four fingers. In this case, it is easy to know the positions of finger joint areas. Then, key point candidates can be further detected in these finger joint areas. In our method, we draw many straight lines to detect finger joint areas, which may result in detecting several different key point candidates in one finger joint area. Thus, a lot of key point candidates may be obtained in four finger joint areas. We then exploit k -means clustering algorithm to calculate four cluster centers, which are treated as the final four key points. Furthermore, utilizing the distance information among four key points, we can know the positions of P_2 and P_4 key points, which can be used to construct coordinate system. Finally, in this new coordinate system, after rotation normalization, the ROI can be extracted from the central region of hand.

The contributions of this paper are as follows:

- We propose a novel method to extract palmprint ROI from the whole hand image. The proposed method first exploits straight line clusters to detect the finger joint

areas, then further detect the key points. To the best of our knowledge, it is the first time to use such strategy to extract palmprint ROI. Thus, our method is very novel.

- The proposed method has several special advantages: (1) Unlike tangent-based method, the proposed method does not need to know approximate positions of finger joint areas in advance. (2) Unlike the contour profile distance distribution-based method, the proposed method does not need to set reference point in advance. (3) The proposed method does not need to extract the contour of the hand. Thus, it will not be affected by noise in contour. (4) The strategy of using dense straight line clusters can well detect the finger joint areas, which makes the proposed method very robust. (5) The strategy of using k -means clustering algorithm to determine four key points makes the positions of four key points more stable. (6) The proposed method can judge the image is captured from left or right hand, which is an important information and is very useful for following feature extraction. (7) The proposed method needs few parameters.

II. RELATED WORK

In the field of palmprint recognition research, the tangent-base method and the contour profile distance distribution-based method are two main ROI extraction methods. In this section, we will briefly revisit them and then present our motivation.

A. TANGENT-BASED METHOD

Zhang et al. [30] designed a contact palmprint acquisition device as shown in Fig. 1(a), and proposed an effective palmprint ROI extraction algorithm. This algorithm first applies a lowpass filter, such as Gaussian smoothing, to the original image (see Fig. 4(a)). Then, a threshold is used to convert the convolved image to a binary image, as shown in Fig. 4(b). Because the acquisition device uses two pegs to fix the hand, it is easy to know the position of finger joint areas of R_2 and R_4 . The boundaries of R_2 and R_4 can be obtained by a boundary tracking algorithm, as shown in Fig. 4(c). And then, the tangent of the two gaps of R_2 and R_4 can be computed to determine the key points P_2 and P_4 , as shown in Fig. 4(d). At last, line up P_2 and P_4 to form a new coordinate system (see Fig. 4(e)), and crop the ROI subimage (see Fig. 4(f)), which is located at a certain area of the hand image.

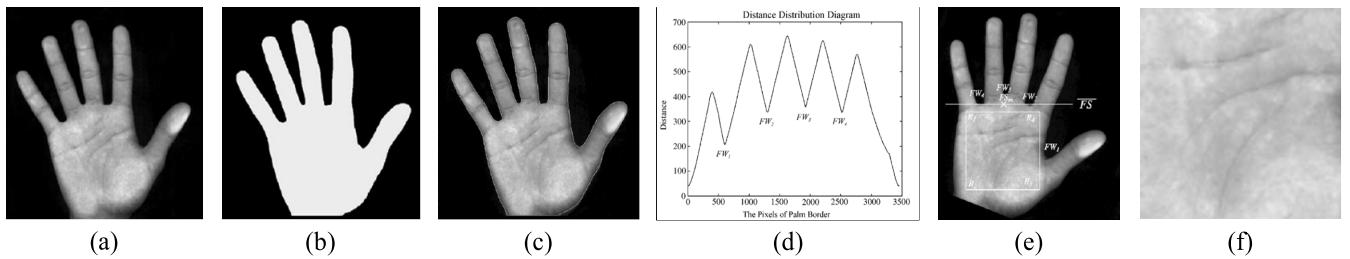


FIGURE 5. The main steps of the contour profile distance distribution-based algorithm [31]. (a) original hand image, (b) binary image, (c) the boundaries of the hand, (d) the contour profile distance distribution, (e) the new coordinate system in hand image, and (f) final ROI image.

After tangent-based method was proposed, it has become one of the most important methods for palmprint ROI extraction. Recently, Zhang *et al.* [33] designed a contactless palmprint acquisition device, and they also exploited tangent-based method for contactless palmprint ROI extraction.

B. CONTOUR PROFILE DISTANCE DISTRIBUTION-BASED MEHTOD

The contour profile distance distribution-based method is another important method for palmprint ROI extraction, which is first proposed by Lin *et al.* [31]. They used a scanner to capture the whole hand image, as shown in Fig. 5(a). The main steps of this method are as follows. First, a traditional median filter is utilized to reduce the noise of captured image. Next, the binary image is obtained by an adaptive thresholding (see Fig. 5(b)). Then, the boundaries of whole hand are obtained using a boundary tracking algorithm (see Fig. 5(c)). In the binary hand image, there is an intersection line between the wrist and the bottom margin of hand. The middle point of the intersection line is treated as the reference point. Calculate the Euclidean distance between the reference point and all pixels of the hand boundary. In this way, the curve of contour profile distance distribution can be obtained (see Fig. 5(d)). In this curve, there are five local maximums and four local minimums. Obviously, four local minimums correspond to four key points P_1 , P_2 , P_3 , and P_4 . At last, line up P_2 and P_4 to form a new coordinate system (see Fig. 5(e)), and crop the ROI subimage (see Fig. 5(f)).

C. OTHER IMPORTANT PALMPRINT ROI EXTRACTION METHODS

Besides the tangent-based method [30] and the contour profile distance distribution-based method [31], there are several other palmprint ROI extraction methods.

In order to detect the key points from four finger joint areas, Michael *et al.* [36] proposed a key points detection method similar to the methods of corner detection. In [36], a binary hand image was first obtained after segmentation and thresholding. Then, a competitive hand valley detection (CHVD) algorithm was proposed to locate the ROI of the palm. In CHVD algorithm, the contour of the hand is obtained by a using a boundary tracking algorithm. In the

contour of the hand, a pixel was considered a valley if it has some neighboring points lying in the non-hand region while the majority neighboring points are in the hand region. At the same time, this pixel must satisfy all the four predefined conditions, in this way, it can be qualified as a valley location. After the four valleys were detected, the key points P_2 and P_4 can be determined. Line up P_2 and P_4 to form a new coordinate system, and then crop the ROI subimage.

In those hand images with complex background, it is difficult to exactly segment hand region from complex background, which will lead to a difficulty of high-precision binarization. To solve this problem, Aykut and Ekingci [37] proposed a model-based ROI extraction algorithm. In their algorithm, the segmentation of the hand was realized by a model-based segmentation method named as Active Appearance Model (AAM) method. After palm image has been segmented, a regression-based approach was utilized for the extraction of ROI which utilizes the hand shape model.

Recently, Lin *et al.* [38] proposed an inscribed circle-based ROI extraction algorithm, in which the ROI region is located and extracted based on the maximum inscribed circle and centroid methods.

D. MOTIVATION OF OUR WORK

Although aforementioned ROI extraction methods have shown their effectiveness, they still have some limitations. For example, inscribed circle-based ROI extraction algorithm has one shortcoming, that is, it sometimes cannot accurately detect the maximum inscribed circle of the palm, which will lead to poor performance of ROI extraction. Model-based method has two weak points: needing training and high computational complexity. CHVD method is sensitive to the variations of hand pose. Tangent-based algorithm need to know the approximate position of position of finger joint areas of R₂ and R₄ in advance, thus, it is not suitable for extracting palmprint ROI from whole hand image.

The contour profile distance distribution-based method is the most popular ROI extraction method, which can also be used to extract ROI from whole hand image. In the contour profile distance distribution-based method, a reference point should be given in advance to calculate the contour profile distance distribution. However, because different image qualities, sometimes the reference point cannot be correctly

determined, and sometimes the contour profile of hand may have many noises, which will lead to the failure of ROI extraction.

In this paper, we propose a novel method to extract palmpoint ROI from whole hand image. The motivation of our method is that because the fingers have distinctive appearance, we can detect them first. Then, we can know finger joint areas according to fingers' position. At last, four key points can be detected in four finger joint areas. Here, a problem is that how to detect the fingers. A priori knowledge is that there are several intervals between fingers in the whole hand image. Therefore, we can use straight line clusters to detect fingers. Here, it should be noted that the proposed method can only extract ROI from the whole hand images with clean background.

III. WHOLE HAND IMAGE DATABASE

We have designed a contact palmpoint acquisition device as shown in Fig. 6, which can conveniently capture whole hand image. The captured database is called as Hefei University of Technology (HFUT) hand database. The HFUT database is collected in the condition of natural light and by peg-free capturing manner. And, the database is collected from 800 hands corresponding to 400 individuals. For each hand, the images are captured in two different sessions with an interval of one week, where 10 samples were captured in the first session and the second session, respectively. Therefore, the total number of hand images is 16,000. The size of each image is 1000×1000 , and the background of images is black. Fig. 7 shows 8 hand images in HFUT database, in which four images in the first row are captured from the right hands, and four images in the second row are captured from the left hands.



FIGURE 6. A contact hand image acquisition device designed by our team.

IV. METHODOLOGY

A. HAND IMAGE PREPROCESSING

In our method, the first operation is image preprocessing including image segmentation and binarization. To this end,

we apply a lowpass filter, Gaussian smoothing, to the original image I to remove noise. Then, the OTSU algorithm is used to convert the convolved image to a binary image. In order to further remove noise, only the largest connected component is preserved in the binary image. In this way, we can obtain the final binary image B .

B. DETECT THE KEY POINTS IN FINGER JOINT AREAS USING STRAIGHT LINE CLUSTERS

1) EQUATION OF STRAIGHT LINES

In the second stage of our method, we will draw straight line clusters according to several predefined rules to detect finger joint areas, and further to detect the key points. In our method, the equation of straight lines is presented by the point-slope form, as follows:

$$y = k_m(x - x_n) + y_n \quad (1)$$

where k_m represents different slopes, and (x_n, y_n) means different points that the straight lines will pass through.

In this paper, the Eq. (1) can describe two kinds of straight lines. The first one is parallel line cluster passing through different points, and the second one is ray cluster passing through one point.

2) DETECT THE KEY POINT CANDIDATES USING VERTICAL LINE CLUSTER

In HFUT hand database, the direction of hand is left toward, thus, vertical lines can be first used to detect the finger joint areas. Here, the equation of vertical lines is presented as follows:

$$y = (x - x_n) + y_n \quad (2)$$

Obviously, the slope of vertical lines is 1. Meanwhile, these vertical lines are parallel lines and pass through different points (x_n, y_n) .

The predefined rule to draw vertical lines is described as follows:

The predefined rule to draw vertical lines: In the binary image B , we first find the bounding box of hand region. An example is given in Fig. 9(a), in which the yellow box is the bounding box of hand region. Then, we draw the first vertical line from the starting point. Fig. 9(a) shows the first starting point (x_1, y_1) to draw a vertical line, which is a red point. In our method, the position of the first starting point is in the place of $1/3$ to the right of the upper boundary. Next, we continue to draw other vertical lines from right to the left at an interval of 40 pixels until the boundary of bounding box, as shown in Fig. 9(b).

When each vertical line is drawn, we use **Algorithm 1** to detect the key point candidates. First, we draw single straight line L_i in binary hand image B , and count the number N of intersection points between L_i and the hand region in B as shown in Fig. 10(a). If there are 8 or more intersection points ($N \geq 8$) between straight line and hand region, we can conclude that this line passes through four fingers.

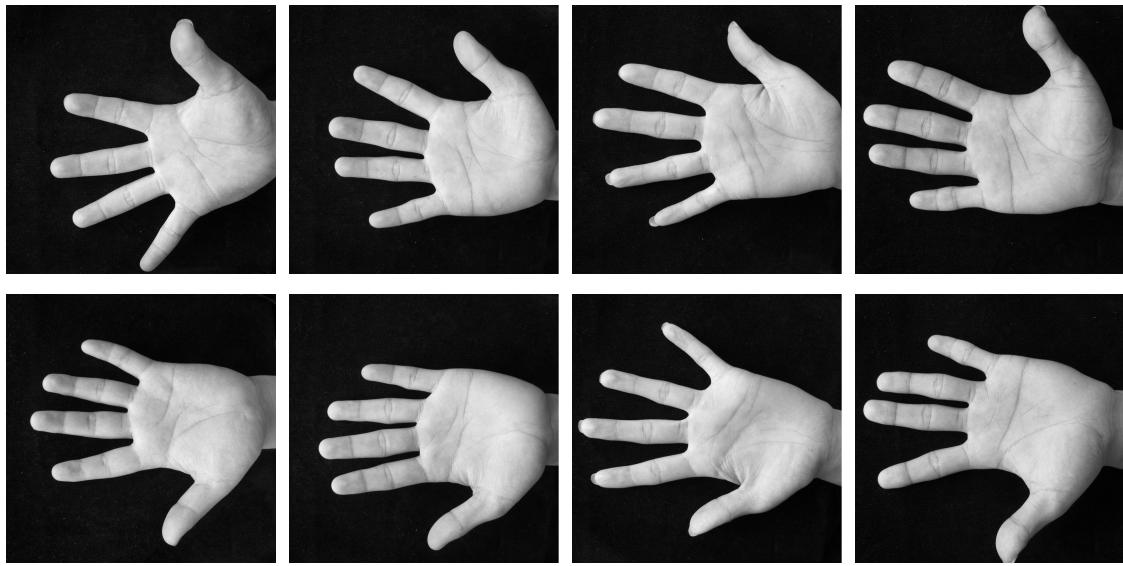


FIGURE 7. 8 hand images in HFUT database, four images in the first row are captured from the right hand, and four images in the second row are captured from the left hand.

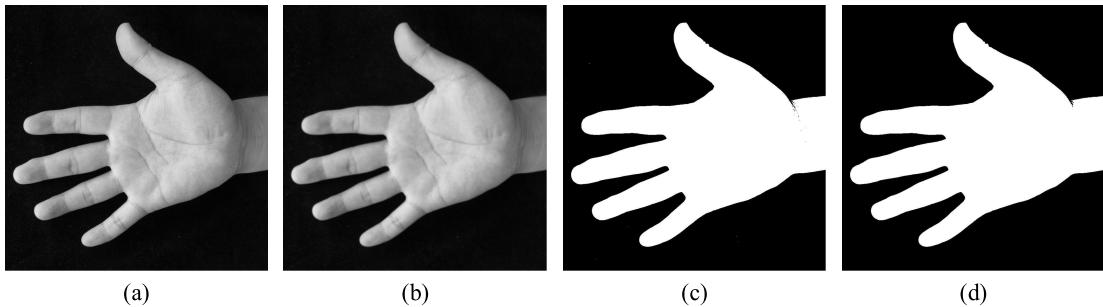


FIGURE 8. The main steps of hand image preprocessing. (a) Original hand image I , (b) the hand image after Gaussian smoothing, (c) binary image, and (d) binary image B after preserving the largest connected component.

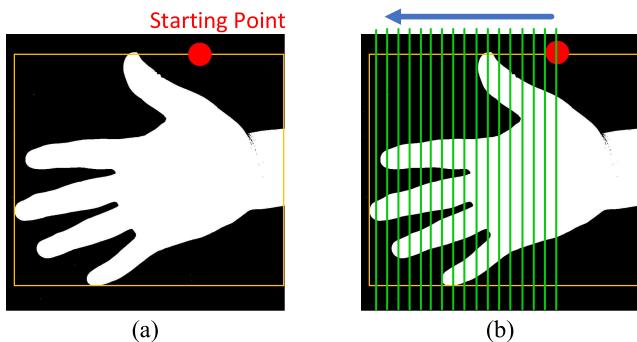


FIGURE 9. The predefined rule to draw vertical lines. (a) The bounding box (yellow box) of hand and the first starting point to draw vertical lines and (b) draw other vertical lines (green lines) from right to the left at an interval of 40 pixels.

In this case, it is easy to know the positions of finger joint area R_2 , and we can first detect the key points candidates in R_2 . To this end, the segment between the second and third intersection points is treated as the left boundary of finger joint area R_2 , as shown in Fig. 10(b) and (c). Then, we use

“Region Growing” algorithm to traverse all pixels of the finger joint area R_2 , the rightmost point C_{Li}^{R2} is recorded as the key point candidate of R_2 , as shown in Fig. 10(d). Add new key point candidate C_{Li}^{R2} to the queue Q , which contains all key point candidates. In the similar way, we can add new key point candidates in R_3 , R_4 , and R_1 to Q . Figure 11 shows the result of using multiple vertical lines to detect key points.

3) DETECT THE KEY POINT CANDIDATES USING RAY CLUSTER

Because the directions of hand may have some changes in different images, only using the vertical lines may not well detect the key point candidates in four finger joint areas. Therefore, our method tries to draw more straight line clusters for robust key point candidates detecting. To this end, we draw ray clusters with different directions from different starting points. Here, the function of ray clusters passing through one point (x_0, y_0) with different slopes of k_m is presented as follows:

$$y = k_m(x - x_0) + y_0 \quad (3)$$

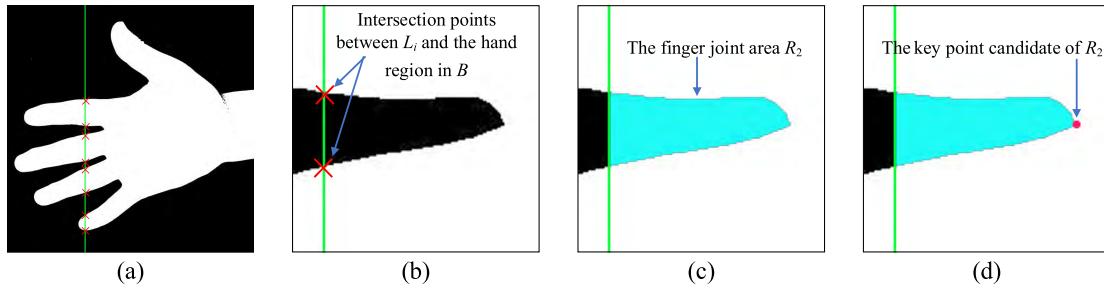


FIGURE 10. Using one vertical line to detect key point candidate. (a) One vertical line passing through four fingers, (b) one vertical line passing through finger joint area R_2 , (c) using region growing algorithm to traverse all pixels of R_2 , and (d) detected key point in R_2 .

Algorithm 1 Using a single straight line to detect the key point candidates

Input: The binary hand image B

Output: The queue Q containing all key point candidates

Step1: Draw a single straight line L_i in binary hand image B .

Step2: Count the number N of intersection points between L_i and the hand region in B .

Step3: If $N \leq 8$

then quit and draw next straight line.

else go to Step 4

End

Step4: The segment between the second and third intersection points is treated as the left boundary of finger joint area R_2 .

Step5: In the right part of the segment, using “Region Growing” algorithm to traverse all pixels of the finger joint area R_2 .

Step6: In the finger joint area R_2 , the rightmost point C_{Li}^{R2} is recorded as the key point candidate of R_2 . Add new key point candidate C_{Li}^{R2} to the queue Q .

Step7: Using the similar operations of Steps 4, 5, and 6 to detect the key point candidates C_{Li}^{R3} , C_{Li}^{R4} , C_{Li}^{R1} of R_3 , R_4 and R_1 . Add new key point candidate C_{Li}^{R3} , C_{Li}^{R4} , C_{Li}^{R1} to the queue Q .

Step8: Return the queue Q

In our method, we set six starting points to draw ray clusters. Fig. 12 shows these six starting points, which are red points. Three starting points locate in the top border of the bounding box, and other three starting points locate in the bottom border of the bounding box. The positions of these six starting points are given as follows:

Starting point 1 is the upper left vertex of the bounding box.

Starting point 2 is the point locating in 1/4 of the top border in the left.

Starting point 3 is the upper right vertex of the bounding box.

Starting point 4 is the lower left vertex of the bounding box.

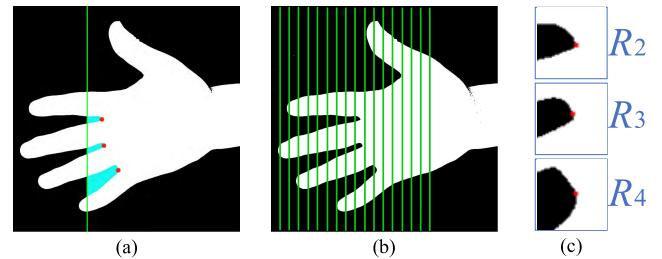


FIGURE 11. Using vertical lines to detect key points. (a) Using one vertical line to detect key point candidates, (b) Using vertical line cluster to detect key point candidates, and (c) the final detected key point candidates using vertical line cluster.

Starting point 5 is the point locating in 1/4 of the bottom border in the left.

Starting point 6 is the lower right vertex of the bounding box.

The predefined rules to draw clusters from six starting points are presented as follows:

The predefined rule to draw ray clusters from Starting point 1 (see Fig. 12(a)): We draw these rays from top to bottom. The first ray across starting point 1 is a vertical line. The interval of end point of other rays is 40 pixels. And the position of the end point of the last ray is near to the lower right vertex.

The predefined rule to draw ray clusters from Starting point 2 (see Fig. 12(b)) : We draw these rays from top to bottom. The first ray across starting point 2 is a vertical line. The interval of end point of other rays is 40 pixels. And the position of the end point of the last ray is near to the lower right vertex.

The predefined rule to draw ray clusters from Starting point 3 (see Fig. 12(c)) : We draw these rays from top to bottom. The first ray across starting point 3 is a vertical line. The interval of end point of other rays is 40 pixels. And the position of the end point of the last ray is near to the lower right vertex.

The predefined rule to draw ray clusters from Starting point 4 (see Fig. 12(d)) : We draw these rays from bottom to top. The first ray across starting point 4 is a vertical line. The interval of end point of other rays is 40 pixels. And the

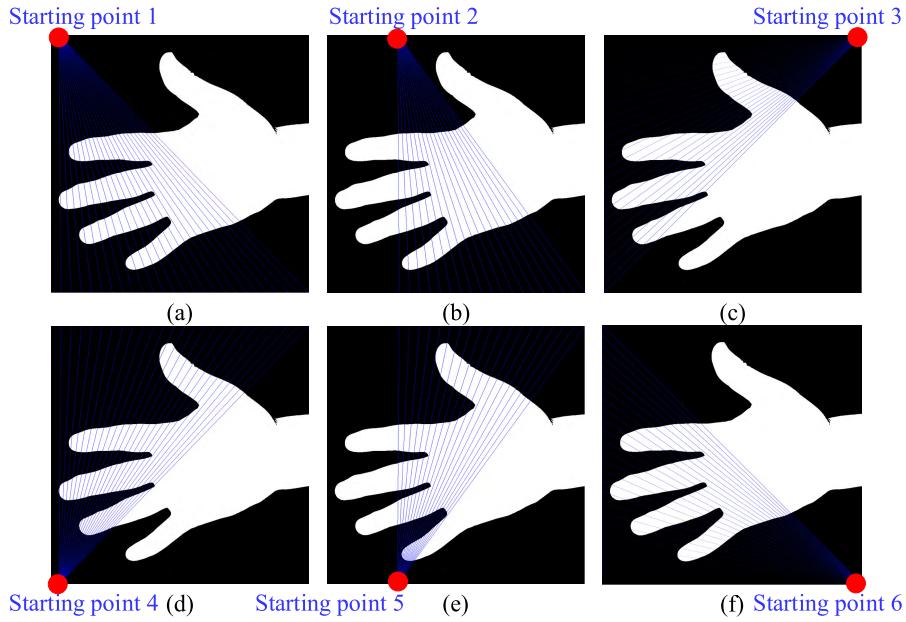


FIGURE 12. Different ray clusters from different starting points. (a) Ray clusters from starting point 1, (b) ray clusters from starting point 2, (c) ray clusters from starting point 3, (d) ray clusters from starting point 4, (e) ray clusters from starting point 5, and (f) ray clusters from starting point 6.

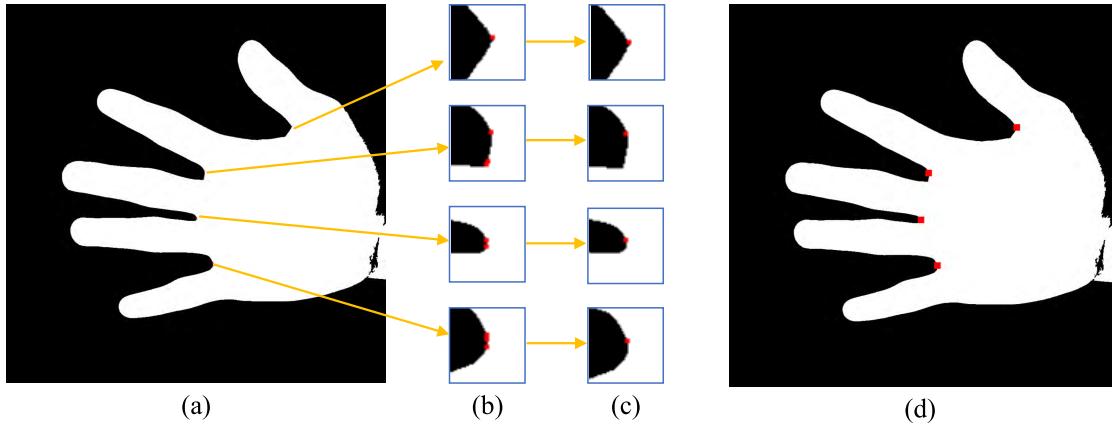


FIGURE 13. Using K-means clustering algorithm to calculate the final key points. (a) the hand image after key points detection based on straight line clusters, (b) there are multiple key points in the finger joint areas, (c) the final key points after K-means clustering, and (d) hand image with final four key points.

position of the end point of the last ray is near to the upper right vertex.

The predefined rule to draw ray clusters from Starting point 5 (see Fig. 12(e)) : We draw these rays from bottom to top. The first ray across starting point 5 is a vertical line. The interval of end point of other rays is 40 pixels. And the position of end point of the last ray is near to the upper right vertex.

The predefined rule to draw ray clusters from Starting point 6 (see Fig. 12(f)) : We draw these rays from bottom to top. The first ray across starting point 6 is a vertical line. The interval of end point of other rays is 40 pixels. And the position of end point of the last ray is near to the upper left vertex.

In these ray clusters, for each straight line, we can detect key point candidates within four finger joint areas using Algorithm 1. After the detection has been conducted using all straight line clusters, the final queue Q containing all key point candidates can be obtained.

4) USE K-MEANS CLUSTERING ALGORITHM TO DETERMINE THE FINAL FOUR KEY POINTS

In our methods, we use different straight lines to detect key points, which may lead to detect several different key point candidates in one finger joint area. An example is given in Fig. 13, Fig. 13(a) is an image after key points detection. In its finger joint areas as shown in Fig. 13(b), it can be seen that there are several key point candidates in four finger

joint areas. We then exploit k -means clustering algorithm to calculate four clustering centers (see Fig. 13(c)). In k -means clustering algorithm, we input the queue Q and set the number of clustering centers to 4. The four clustering centers are treated as the final four key points (see Fig. 13(d)).

C. DETERMINE KEY POINTS OF P_2 AND P_4

In HFUT palmprint database, half of images were captured from right hands as shown in Fig. 14(a), and the other half of images were captured from left hands as shown in Fig. 14(b). After detecting four key points in four finger joint areas, it is needed to know which two points are P_2 and P_4 . Thus, we should judge that each hand image is captured from left hand or right hand. Fortunately, the distance relationship among four key points can help to solve this problem. Because key point P_1 is far away other three key points, we can use this priori knowledge to first determine which key points is P_1 . For two key points P_i and P_j , their corresponding coordinates are (x_i, y_i) and (x_j, y_j) , and Euclidean distance $d_{i,j}$ between P_i and P_j can be calculated using the following equation:

$$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (4)$$

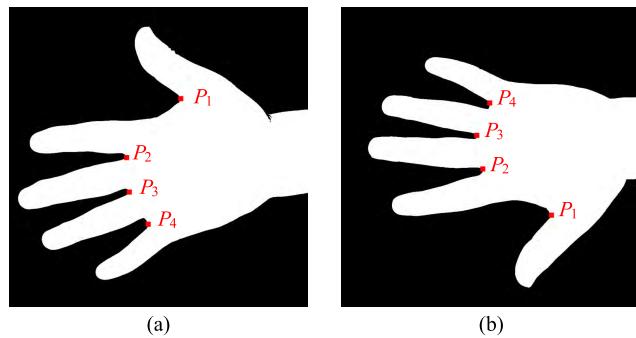


FIGURE 14. Four key points in left hand and right hand. (a) Four key points in right hand and (b) four key points in left hand.

Among four key points, for an arbitrary key point, the sum of its distances from other three key points, D_i , can be calculated by Eq. (5).

$$D_i = \sum_{j=1}^4 d_{i,j} (j=1, 2, 3, 4) \quad (5)$$

After the values of $D_i (i = 1, 2, 3, 4)$ have been calculated, we select the largest value of D_i to determine the key point of P_1 using Eq. (6):

$$P_1(i) = \text{argMax}(D_i) \quad (6)$$

Next, according to the distance away from P_1 , we can determine the key points of P_2 , P_3 , and P_4 in turn.

D. ROTATION NORMALIZATION AND ROI EXTRACTION

In this stage, we line up key points P_2 and P_4 to form a line segment P_2P_4 . An example is given in Fig. 15(a), in which

the red line segment is P_2P_4 . Then, we need to calculate the intersection angle θ between P_2P_4 and the vertical line. Suppose that the spatial coordinates of P_2 and P_4 are (x_2, y_2) and (x_4, y_4) , the intersection angle θ can be calculate by the arctangent function:

$$\theta = \arctan\left(\frac{y_4 - y_2}{x_4 - x_2}\right) \times \left(\frac{180}{\pi}\right) \quad (7)$$

In the new coordinate, the midpoint of segment P_2P_4 is treated as the origin of coordinate O , which is the yellow point in Fig. 15(a). According to intersection angle θ , we rotate the hand image around O to make P_2P_4 become a vertical line. Fig. 15(b) is the hand image after rotation. In the rotation normalization image, a square region, whose center locates in the X-axis, is selected as the ROI subimage, as shown in Fig. 15(c). Fig. 15(d) shows the cropped ROI subimage.

V. EXPERIMENTAL RESULTS

In this section, two methods are used to verify the effectiveness of the proposed method. The first one is manual checking. The second one is “finding the most dissimilar ROI pairs within one class”.

A. EXPERIMENTAL RESULTS BY MANUAL CHECKING

As the name implies, this method is to verify whether the ROI images are correctly extracted by manual checking. Meanwhile, we also conduct the ROI extraction by the method of contour profile distance distribution for performance comparison. As we know, in the method of contour profile distance distribution, a reference point should be given in advance. Usually, the reference point is the midpoint of wrist’s contour. An example is given in Fig. 16, in which the red point is the reference point. However, because different image qualities, sometimes the contour profile of hand may have many noises as shown in Fig. 16, which lead to the failure of ROI extraction. In Table 1, we list the accuracy of ROI extraction in HFUT database. It can be seen that our method achieves the accuracy of 100%, while the accuracy of the algorithm of contour profile distance distribution is 99.325%.

TABLE 1. The ROI extraction accuracy of different methods on HFUT database.

Method	The number of accurate ROI extraction	Accuracy rate
Our method	16000	100%
The method of contour profile distance distribution	15892	99.325%

In order to further show the effect of the proposed algorithm, Fig. 17 depicts four ROI subimages, which were extracted from four hand image of the first class in HFUT database. These four hand images include: the first and tenth hand images captured in the first session, and the eleventh and twentieth hand images captured in the second session. From Fig. 17, it can be seen our method can extract ROI very well. Fig. 18 shows the 8 ROI subimages extracted from 8 hand

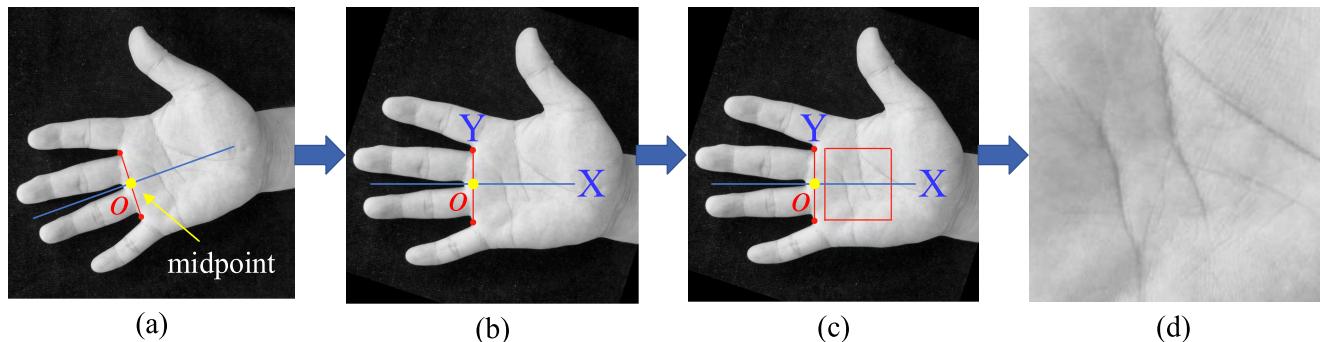


FIGURE 15. Rotation normalization and ROI extraction. (a) Line up P_2 and P_4 to form a segment P_2P_4 , (b) rotate the segment P_2P_4 to make it become a vertical line, (c) a square region is selected as the ROI subimage, and (d) the extracted ROI image.

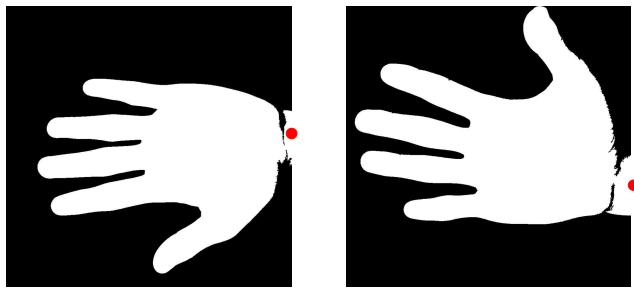


FIGURE 16. The contour profile of hand may have many noises, the red point is the reference point.

images of Fig. 7, it can be seen that these ROI subimages can well contain the palmprint region.

B. EXPERIMENTAL RESULTS OF “FINDING THE MOST DISSIMILAR ROI PAIRS WITHIN ONE CLASS”

To further verify the effectiveness of the proposed algorithm, we exploit the strategy of “finding the most dissimilar ROI pairs within one class”, that is, we try to find the most dissimilar ROI pairs within one class by the recognition methods, and then to see whether the dissimilarity of this ROI pairs is caused by our ROI extraction method. In this paper, we select the methods of Competitive Code [30] and Robust Line Orientation Code (RLOC) [39] to conduct the recognition task. Meanwhile, Competitive Code and RLOC are two classical recognition methods, and are all coding-based palmprint recognition methods. In Competitive Code, six real parts of 2D ellipsoidal Gabor filters with different directions are used to convolute an image to obtain six Gabor response. Then, the index number of the minimum line response of each pixels is used to construct an orientation representation image. Since there are totally 6 different orientations, each orientation is encoded by 3 bits. At last, the difference between two direction maps is measured by the Hamming distance. In RLOC, the modified finite Radon transform (MFRAT) with six directions is exploited to extract the orientation feature of palmprint. And the orientation map is matched by pixel-to-area comparison.

For two arbitrary ROI pairs, their matching score of Competitive Code is in the range of [0, 1]. In Competitive Code, if the matching score is close to 1 it means that two palmprint are more dissimilar. For two arbitrary ROI pairs, their matching score of RLOC is also in the range of [0, 1].

However, in RLOC, if the matching score is close to 0, it means that two palmprint are more dissimilar.

In HFUT database, for each palm, the first image is used for training, and the rest 19 images are used for test. In this paper, both verification and identification experiments are conducted. Here, we firstly introduce some measures, for performance evaluation. For identification, Accurate Recognition Rate (ARR) is exploited to evaluate the identification performance, which is rank 1 identification rate. In verification experiments, the statistical pairs of false rejection rate (FRR) and false acceptance rate (FAR) were adopted to evaluate the performance of our approach. To obtain the statistical pairs of FRR and FAR, each of the test images was matched with all of the training templates. If the test palmprint image and the template are from the same palm, the matching between them is remarked as a genuine matching. Likewise, an impostor matching can also be defined in a similar manner. For verification experiments, equal error rate (EER) is used as performance measure where the FAR and FRR are equal. Figs. 19(a) and (b) show the distributions of the genuine and impostor matching scores obtained from Competitive Code and RLOC. Table 2 lists the recognition performance of Competitive Code and RLOC on HFUT database.

TABLE 2. The recognition performance of different methods on HFUT database.

Recognition methods	ARR	EER
Competitive Code	99.6461%	0.3461%
RLOC	99.5066%	0.4641%

From the definition of FRR, we know that FRR is a statistic of intra-class matching scores. As we have mentioned above, for the method of Competitive Code, if the matching score of a pair of palmprint is close to 1, it means that two palmprint are more dissimilar. In all intra-class matching

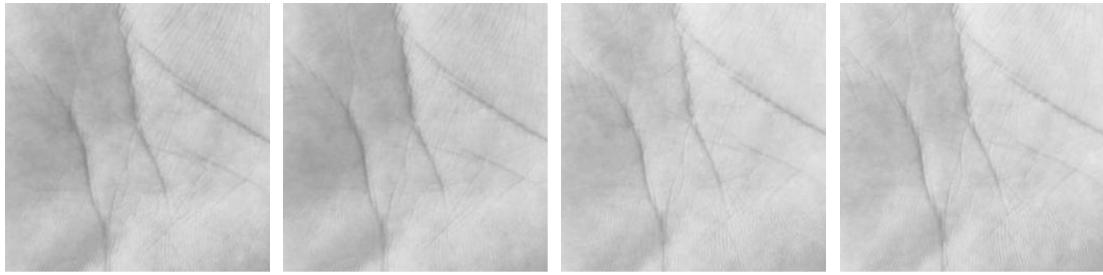


FIGURE 17. 4 ROI subimage extracted from 4 hand images of the first class in HFUT database.

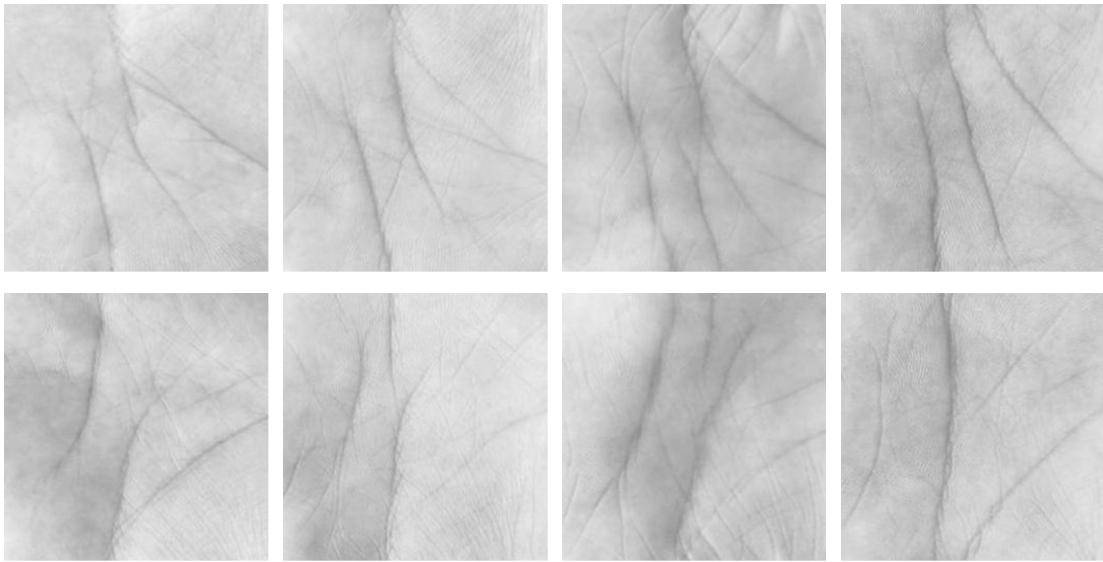


FIGURE 18. 8 ROI subimages extracted from 8 different images captured from 8 different hands.

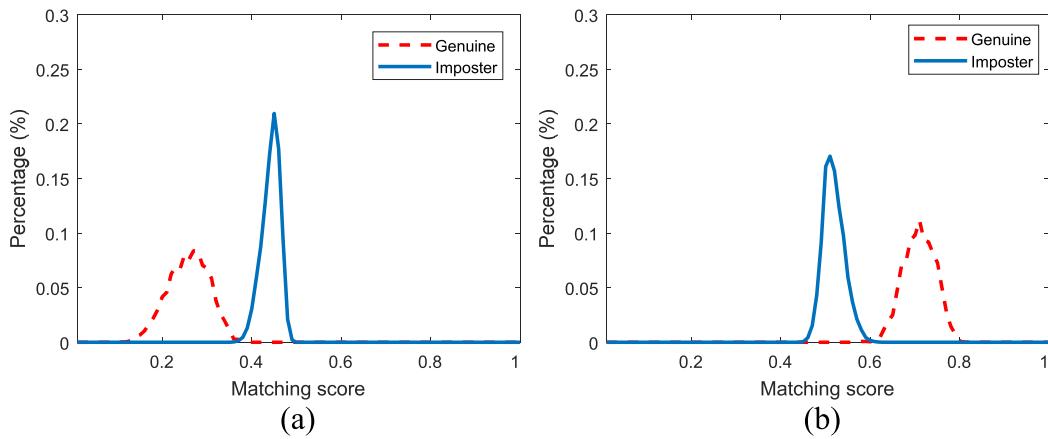


FIGURE 19. The distributions of the genuine and impostor matching scores of different recognition methods.
(a) Competitive code and (b) RLOC.

scores, we find the largest intra-class matching score, which is 0.4367, and we show the corresponding image pairs in Fig. 20(a). Although the matching score between two ROI images in Fig. 20(a) is highest, two ROI images have been well extracted by our method. That is to say, the dissimilarity

of two ROI image is not caused by our ROI extraction method. For the method of RLOC, if the matching score of a pair of palmprint is close to 0, it means that two palmprint are more dissimilar. In all intra-class matching scores, we find the smallest intra-class matching score, which is 0.5107, and we

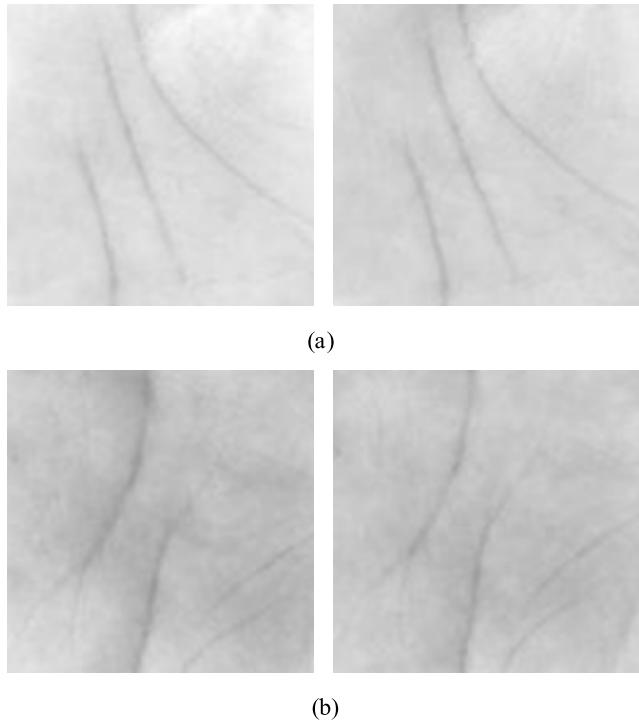


FIGURE 20. The most dissimilar ROI pairs within one class searched by different recognition methods. (a) The most dissimilar ROI pairs within one class searched by Competitive Code. (b) The most dissimilar ROI pairs within one class searched by RLOC.

show the corresponding image pairs in Fig. 20(b). Although the matching score between two ROI images in Fig. 20(b) is lowest, two ROI images have also been well extracted by our method, which shows the robustness of our method.

VI. CONCLUSION

In this paper, we proposed a novel palmprint ROI extraction method, which can extract palmprint ROI from whole hand images using straight line clusters. In our method, in binary hand image, dense straight lines were drawn to first detect the region of fingers. After this detection, it is easy to know the positions of finger joint areas. We then used the “Region Growing” algorithm to traverse all pixels in the finger joint areas, and the rightmost points in finger joint areas were recorded as the key point candidates. Next, k -means clustering algorithm was exploited to calculate four cluster centers of all key point candidates, which are treated as the final four key points. Furthermore, utilizing the distance information among four key points, the position of P_2 and P_4 key points were determined, which were used to construct a new coordinate system. Finally, in this new coordinate system, after rotation normalization, the ROI was extracted from the central region of hand. The core idea of our method is totally different with other ROI extraction methods. Thus, our method is very novel. And, the proposed method has several special advantages. For example, it does not need to know approximate positions of finger joint areas in advance; it does not need to set reference point in advance; it does not need to extract the contour of the hand; and it needs few parameters.

Thus, the proposed method has strong applicability. Furthermore, the strategy of using dense straight line clusters to detect can well detect the finger joint areas, which makes the proposed method very robust. And the strategy of using k -means clustering algorithm to determine four key points makes the four key points more stable. In this work, we collected a database including 16000 whole hand images, which is a large hand image database in the field of palmprint recognition. In this database, experimental results demonstrate that the proposed method can achieve 100% localization and extraction accuracy, which verified the effectiveness of the proposed method. The proposed method has several disadvantages: (1) the speed of the proposed method is slow; (2) the proposed method is only suitable for extracting ROI from the whole hand images with clean background, which may be influenced by the image quality; (3) the proposed method cannot extract ROI from hand with arbitrary direction. In the future work, we plan to add an image quality assessment module [40], and we will further improve the proposed method so that it can extract ROI from hand with arbitrary direction.

REFERENCES

- [1] A. K. Jain, A. Ross, and S. Prabhakar, “An introduction to biometric recognition,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 4–20, Jan. 2004.
- [2] A. K. Jain, K. Nandakumar, and A. Ross, “50 years of biometric research: Accomplishments, challenges, and opportunities,” *Pattern Recognit. Lett.*, vol. 79, pp. 80–105, Aug. 2016.
- [3] B. Jiang, J. Yang, Z. Lv, and H. Song, “Wearable vision assistance system based on binocular sensors for visually impaired users,” *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1375–1383, Apr. 2019. doi: [10.1109/JIOT.2018.2842229](https://doi.org/10.1109/JIOT.2018.2842229).
- [4] A. Kong, D. Zhang, and M. Kamel, “A survey of palmprint recognition,” *Pattern Recognit.*, vol. 42, no. 7, pp. 1408–1418, 2009.
- [5] L. Fei, G. Lu, W. Jia, S. Teng, and D. Zhang, “Feature extraction methods for palmprint recognition: A survey and evaluation,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 2, pp. 346–363, Feb. 2019.
- [6] D. Zhang, W. Zuo, and F. Yue, “A comparative study of palmprint recognition algorithms,” *ACM Comput. Surv.*, vol. 44, no. 1, 2012, Art. no. 2.
- [7] J. Rodríguez-Ruiz, M. A. Medina-Pérez, R. Monroy, and O. Loyola-González, “A survey on minutiae-based palmprint feature representations, and a full analysis of palmprint feature representation role in latent identification performance,” *Expert Syst. Appl.*, vol. 131, pp. 30–44, Oct. 2019.
- [8] D. Zhong, X. Du, and K. Zhong, “Decade progress of palmprint recognition: A brief survey,” *Neurocomputing*, vol. 328, pp. 16–28, Feb. 2019.
- [9] P. Chen, B. Ding, H. Wang, R. Liang, Y. Zhang, W. Zhu, and Y. Liu, “Design of low-cost personal identification system that uses combined palm vein and palmprint biometric features,” *IEEE Access*, vol. 7, pp. 15922–15931, 2019.
- [10] F. Hao, L. Yang, G. Yang, N. Liu, and Z. Liu, “RFPIQM: Ridge-based forensic palmprint image quality measurement,” *IEEE Access*, vol. 6, pp. 62076–62088, 2018.
- [11] I. Rida, S. Al-Maadeed, A. Mahmood, A. Bouridane, and S. Bakshi, “Palmprint identification using an ensemble of sparse representations,” *IEEE Access*, vol. 6, pp. 3241–3248, 2018.
- [12] H. Qin, M. A. El Yacoubi, J. Lin, and B. Liu, “An iterative deep neural network for hand-vein verification,” *IEEE Access*, vol. 7, pp. 34823–34837, 2019.
- [13] S. Zhao, B. Zhang, and C. L. P. Chen, “Joint deep convolutional feature representation for hyperspectral palmprint recognition,” *Inf. Sci.*, vol. 489, pp. 167–181, Jul. 2019.
- [14] K. Zhou, X. Zhou, L. Yu, L. Shen, and S. Yu, “Double biologically inspired transform network for robust palmprint recognition,” *Neurocomputing*, vol. 337, pp. 24–45, Apr. 2019.
- [15] L. Fei, B. Zhang, Y. Xu, W. Jia, J. Wen, and J. Wu, “Precision direction and compact surface type representation for 3D palmprint identification,” *Pattern Recognit.*, vol. 87, pp. 237–247, Mar. 2019.

- [16] F. Ma, X. Zhu, C. Wang, H. Liu, and X.-Y. Jing, "Multi-orientation and multi-scale features discriminant learning for palmprint recognition," *Neurocomputing*, vol. 348, pp. 169–178, Jul. 2019.
- [17] L. Fei, B. Zhang, Y. Xu, D. Huang, W. Jia, and J. Wen, "Local discriminant direction binary pattern for palmprint representation and recognition," *IEEE Trans. Circuits Syst. Video Technol.*, to be published.
- [18] L. Fei, B. Zhang, Y. Xu, Z. Guo, J. Wen, and W. Jia, "Learning discriminant direction binary palmprint descriptor," *IEEE Trans. Image Process.*, to be published.
- [19] A. S. ElSayed, H. M. Ebeid, M. I. Roushdy, and Z. T. Fayed, "Masked SIFT with align-based refinement for contactless palmprint recognition," *IET Biometrics*, vol. 8, no. 2, pp. 150–158, Mar. 2019.
- [20] L. Fei, B. Zhang, S. Teng, and W. Zhang, "Local apparent and latent direction extraction for palmprint recognition," *Inf. Sci.*, vol. 473, pp. 59–72, Jan. 2019.
- [21] Y. Xu, L. Fei, J. Wen, and D. Zhang, "Discriminative and robust competitive code for palmprint recognition," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 2, pp. 232–241, Feb. 2018.
- [22] L. Fei, G. Lu, W. Jia, J. Wen, and D. Zhang, "Complete binary representation for 3-D palmprint recognition," *IEEE Trans. Instrum. Meas.*, vol. 67, no. 12, pp. 2761–2771, Dec. 2018.
- [23] D. Palma, P. L. Montessoro, G. Giordano, and F. Blanchini, "Biometric palmprint verification: A dynamical system approach," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.
- [24] L. Zhang, L. Li, A. Yang, Y. Shen, and M. Yang, "Towards contactless palmprint recognition: A novel device, a new benchmark, and a collaborative representation based identification approach," *Pattern Recognit.*, vol. 69, pp. 199–212, Sep. 2017.
- [25] A. Ungureanu, S. Thavalengal, T. E. Cognard, C. Costache, and P. Corcoran, "Unconstrained palmprint as a smartphone biometric," *IEEE Trans. Consum. Electron.*, vol. 63, no. 3, pp. 334–342, Aug. 2017.
- [26] D.-S. Huang, W. Jia, and D. Zhang, "Palmprint verification based on principal lines," *Pattern Recognit.*, vol. 41, no. 4, pp. 1316–1328, 2008.
- [27] W. Jia, B. Zhang, J. Lu, Y. Zhu, Y. Zhao, W. Zuo, and H. Ling, "Palmprint recognition based on complete direction representation," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4483–4498, Sep. 2017.
- [28] Y.-T. Luo, L.-Y. Zhao, B. Zhang, W. Jia, F. Xue, J.-T. Lu, Y.-H. Zhu, and B.-Q. Xu, "Local line directional pattern for palmprint recognition," *Pattern Recognit.*, vol. 50, pp. 26–44, Feb. 2016.
- [29] W. Jia, R.-X. Hu, Y.-K. Lei, Y. Zhao, and J. Gui, "Histogram of oriented lines for palmprint recognition," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 3, pp. 385–395, Mar. 2014.
- [30] D. Zhang, W.-K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041–1050, Sep. 2003.
- [31] C.-L. Lin, T. C. Chuang, and K.-C. Fan, "Palmprint verification using hierarchical decomposition," *Pattern Recognit.*, vol. 38, no. 12, pp. 2639–2652, 2005.
- [32] W. Jia, R.-X. Hu, J. Gui, Y. Zhao, and X.-M. Ren, "Palmprint recognition across different devices," *Sensors*, vol. 12, no. 6, pp. 7938–7964, 2012.
- [33] L. Zhang, Z. Cheng, Y. Shen, and D. Wang, "Palmprint and palmvein recognition based on DCNN and a new large-scale contactless palmvein dataset," *Symmetry*, vol. 10, no. 4, p. 78, 2018.
- [34] B. Jiang, J. Yang, Q. Meng, B. Li, and W. Lu, "A deep evaluator for image retargeting quality by geometrical and contextual interaction," *IEEE Trans. Cybern.*, to be published. doi: [10.1109/TCYB.2018.2864158](https://doi.org/10.1109/TCYB.2018.2864158).
- [35] J. Yang, K. Sim, W. Lu, and B. Jiang, "Predicting stereoscopic image quality via stacked auto-encoders based on stereopsis formation," *IEEE Trans. Multimedia*, to be published. doi: [10.1109/TMM.2018.2889562](https://doi.org/10.1109/TMM.2018.2889562).
- [36] G. K. O. Michael, T. Connie, and A. B. J. Teoh, "Touch-less palm print biometrics: Novel design and implementation," *Image Vis. Comput.*, vol. 26, no. 12, pp. 1551–1560, 2008.
- [37] M. Aykut and M. Ekinici, "Developing a contactless palmprint authentication system by introducing a novel ROI extraction method," *Image Vis. Comput.*, vol. 40, pp. 65–74, Aug. 2015.
- [38] S. Lin, T. Xu, and X. Yin, "Region of interest extraction for palmprint and palm vein recognition," in *Proc. 9th Int. Congr. Image Signal Process., Biomed. Eng. Inform.*, Oct. 2016, pp. 538–542.
- [39] W. Jia, D.-S. Huang, and D. D. Zhang, "Palmprint verification based on robust line orientation code," *Pattern Recognit.*, vol. 41, no. 5, pp. 1504–1513, 2008.
- [40] J. Yang, C. Ji, B. Jiang, W. Lu, and Q. Meng, "No reference quality assessment of stereo video based on saliency and sparsity," *IEEE Trans. Broadcast.*, vol. 64, no. 2, pp. 341–353, Jun. 2018. doi: [10.1109/TBC.2018.2789583](https://doi.org/10.1109/TBC.2018.2789583).



QIANWEN XIAO received the B.Sc. degree in computer science from the Hefei University of Technology, where she is currently pursuing the master's degree with the School of Computer Science and Information Engineering, and also with the Anhui Province Key Laboratory of Industry Safety and Emergency Technology. Her research interests include image processing and biometrics recognition.



JINGTING LU received the B.Sc., M.Sc., and Ph.D. degrees in computer science from the Hefei University of Technology, Hefei, China, in 2004, 2009, and 2014, respectively, where she is currently a Research Assistant Professor with the School of Computer and Information. Her research interests include computer vision, biometrics, pattern recognition, image processing, and machine learning.



WEI JIA received the B.Sc. degree in informatics from Central China Normal University, Wuhan, China, in 1998, the M.Sc. degree in computer science from the Hefei University of Technology, Hefei, China, in 2004, and the Ph.D. degree in pattern recognition and intelligence systems from the University of Science and Technology of China, Hefei, in 2008. He has been a Research Assistant and also an Associate Professor with the Hefei Institutes of Physical Science, Chinese Academy of Sciences, from 2008 to 2016. He is currently a Research Associate Professor with the School of Computer Science and Information Engineering, Hefei University of Technology. He is also a member of the Anhui Province Key Laboratory of Industry Safety and Emergency Technology. His research interests include computer vision, biometrics, pattern recognition, image processing, and machine learning.



XIAOPING LIU received the M.Sc. and Ph.D. degrees from the Department of Computer Science, Hefei University of Technology, China, where he is currently a Professor with the School of Computer Science and Information Engineering. He is also a member of the Anhui Province Key Laboratory of Industry Safety and Emergency Technology. His research interests include computer graphics and cooperative computing.