ISSN No: 2454-9614

Multibiometrics By Combining Left and Right Palmprint Images using SIFT Features

Aishwarya.V.V, Devipriyanga.V*

[a)] *Department Of Electronics and Communication Engineering, Sri Krishna college of Engineering and Technology, Coimbatore , Tamilnadu, India.*

*Corresponding Author: V.V Aishwarya

E-mail: aishwaryasornaraj@gmail.com

Received: 11/11/2015, Revised: 13/12/2015 and Accepted: 07/03/2016

Abstract

The proposed steganography system, where edges in the spread picture have been utilized to insert messages and another piece of plain content is implanted into a sound recording.A RC-4 algorithm is used to encrypt the data. Encrypting data are measured and split into two parts by finding the mid range value using the medium edge detection.From previous research work, to reduce the data losses in the wireless networksthe new methods of integrating quantum cryptography for key distribution were implemented. The Quantum key distribution can be used in wireless networks to securely distribute the keys. The key provides mutual authentication between the two communication ports and send the data in a secure way. Through existingSimulations, the proposed mechanisms achieve significantly anencrypted data transfer is done through a secure communication to, keep the user's Information and data safe when connected through the wireless networks.

Keywords: Palmprint recognition, Multibiometrics, Scale Invariant Feature Transform(SIFT), Radon Transform.

Reviewed by* **ICETSET'16 *organizing committee*

1. Introduction

1.1 Need For Palmprint Technology

Palm print images are captured by acquisition module and are fed into recognition module for authentication.

- Compared with face recognition palm print is hardly affected by age and accessories
- Compared with fingerprint recognition palm print images contain more information and needs only low resolution image capturing devices which reduces the cost of the system.

- Compared with iris recognition the palm print images can be captured without intrusiveness as people might fear of adverse effects on their eyes and cost effective.

The palmprint contains not only principle curves and wrinkles but also rich texture and miniscule points, so the palmprint identification is able to achieve a high accuracy because of available rich information in palmprint. Various palmprint identification methods, such as coding based methods [8] and principle curve methods [5], have been proposed in past decades. In addition to these methods, subspace based methods [2] can also perform well for palmprint identification. For example, Eigenpalm and Fisherpalm [11] are two well-known subspace based palmprint identification methods. In recent years, 2D appearance based methods such as 2D Linear Discriminant Analysis (2DLDA) [2] have also been used for palmprint recognition.

Further, the Representation Based Classification (RBC) method also shows good performance in palmprint identification. Additionally, the Scale Invariant Feature Transform (SIFT) [13], which transforms image data into scale-invariant coordinates, are successfully introduced for the contactless palmprint identification. No single biometric technique can meet all requirements in circumstances.

To overcome the limitation of the unimodal biometric technique and to improve the performance of the biometric system, multimodal biometric methods are designed by using multiple biometrics or using multiple modals of the same biometric trait, which can be fused at four levels: image (sensor) level, feature level, matching score level and decision level.

1.2 Multimodel Biometric System

Multimodal biometric systems use multiple sensors or biometrics to overcome the limitations of unimodal biometric systems. For instance iris recognition systems can be compromised by aging irides and finger scanning systems by worn-out or cut fingerprints [1]. While unimodal biometric systems are limited by the integrity of their identifier, it is unlikely that several unimodal systems will suffer from identical limitations. Multimodal biometric systems can obtain sets of information from the same marker (i.e., multiple images of an iris, or scans of the same finger) or information from different biometrics (requiring fingerprint scans and, using voice recognition, a spoken pass-code).

2.Existing System

2.1 Sift Based Palmprint Recognition Considering Single Palmprint

Many times when images goes under certain transformation deletion of pegs or other auxiliary schemas on image acquisition devices can unavoidably introduce variations of palm print images due to hand unsteadiness. In such cases, this method is used to solve the problem, this method proposes a palm print image alignment method based on SIFT features [13]. SIFT (Scale Invariant feature transform) features are based on local information, which are invariant to image shift, scale, and rotation variations, and partially invariant to illumination and projective changes.

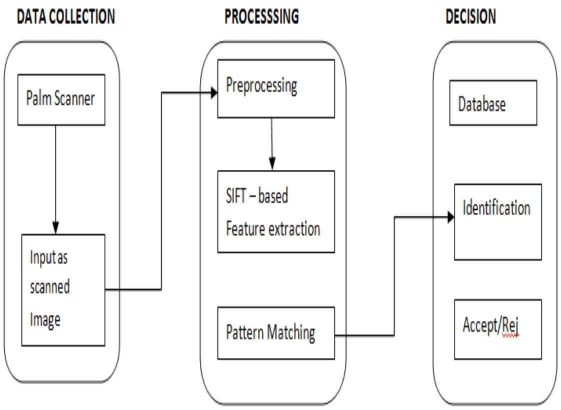


Fig 1 Block diagram of existing system

In SIFT based algorithm, image is first preprocessed using Gabor filter [13]. After preprocessing the image is then for SIFT feature extraction where Scale space construction is performed followed by Key point localization. Orientation assignment is done which is then followed by Descriptor computing. After feature extraction proper alignment is to be done using the Homographic matrix. Given any point in one image, in the homogeneous coordinate system [x, y, 1]^T, the corresponding point in the second image is given by

[x' ,y' ,c] = H [x, y, 1]^T (1)

If the number of correspondences is less than 4the homograph cannot be solved. After image alignment, the query image and the gallery are better aligned.

Inorder to improve the security and the performance of the system it is better to adopt the proposed system.

3.Proposed System

3.1 Combined palmprint based SIFT Palmprint Recognition Method

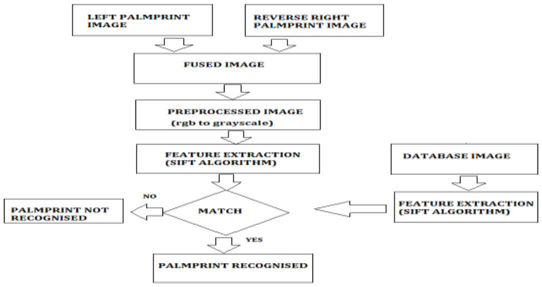

LEFT PALMPRINT IMAGE
REVERSE RIGHT PALMPRINT IMAGE
FUSED IMAGE
PREPROCESSED IMAGE (rgb to grayscale)
FEATURE EXTRACTION (SIFT ALGORITHM)
DATABASE IMAGE
FEATURE EXTRACTION (SIFT ALGORITHM)
NO
MATCH
PALMPRINT NOT RECOGNISED
YES
PALMPRINT RECOGNISED

Fig 2 Block diagram of proposed method

3.2Working principle

Input image

Left palm image is acquired and then reverse right palm image is acquired. Finally the Input image is obtained by fusing left and reversed right palm images which is given as input to the security systems.

Preprocessing

Preprocessing is the process of converting an RGB image into Gray scale image. In addition to that Blurring of the images and noise will be removed using median filter.

Feature Extraction

In this module, features of palm print images are extracted. SIFT and Radon transformation algorithm is used to extract the features.

Database updation

Here, known personspalmprint are stored. Then features of each image is identified using SIFT and Radon transformation.

Matching

Finally the features obtained for Database images and input image is compared. If the features of input image and database image are equal, palm is recognized otherwise not recognized.

3.3 SIFT algorithm description

Scale-invariant feature transform (or SIFT) is an algorithm in computer vision to detect and describe local features in images. It is an image descriptor for image-based matching and recognition. This descriptor as well as related image descriptors are used for a large number of purposes in computer vision related to point matching between different views of a 3-D scene and view-based object recognition. The SIFT descriptor is invariant to translations, rotations and scaling transformations in the image domain and robust to moderate perspective transformations and illumination variations. Experimentally, the SIFT descriptor has been proven to be very useful in practice for image matching and object recognition under real-world conditions.

SIFT keypoints of objects are first extracted from a set of reference images and stored in a database. An object is recognized in a new image by individually comparing each feature from the new image to this database and finding candidate matching features based on Euclidean distance of their feature vectors. From the full set of matches, subsets of keypoints that agree on the object and its location, scale, and orientation in the new image are identified to filter out good matches. The determination of consistent clusters is performed rapidly by using an efficient hash table implementation of the generalized Hough transform. Each cluster of 3 or more features that agree on an object and its pose is then subject to further detailed model verification and subsequently outliers are discarded. Finally the probability that a particular set of features indicates the presence of an object is computed, given the accuracy of fit and number of probable false matches.

The detection and description of local image features can help in object recognition. The SIFT features are local and based on the appearance of the object at particular interest points, and are invariant to image scale and rotation. They are also robust to changes in illumination, noise, and minor changes in viewpoint. In addition to these properties, they are highly distinctive, relatively easy to extract and allow for correct object identification with low probability of mismatch. They are relatively easy to match against a (large) database of local features but however the high dimensionality can be an issue, and generally probabilistic algorithms such as k-d trees with best bin first search are used. Object description by set of SIFT features is also robust to partial occlusion; as few as 3 SIFT features from an object are enough to compute its location and pose. Recognition can be performed in close-to-real time, at least for small databases and on modern computer hardware.

3.4 Different stages of SIFT algorithm

Scale-space extrema detection

The image is convolved with Gaussian filters at different scales, and then the difference of successive Gaussian-blurred images is taken. Keypoints are then taken as maxima/minima of the Difference of Gaussians (DoG) that occur at multiple scales. Specifically, a DoG image is given by $\mathrm{D}(\mathrm{x},\mathrm{y},\sigma)$

$$\mathrm{D}(\mathrm{x},\mathrm{y},\sigma)=\mathrm{L}(\mathrm{x},\mathrm{y},\mathrm{k}\mathrm{i}\sigma)-\ \mathrm{L}(\mathrm{x},\mathrm{y},\mathrm{k}\mathrm{j}\sigma) \qquad (2)$$

where L (x, y, kσ) is the convolution of the original image I(x, y) with the Gaussian blur G at scale G (x, y, kσ) which is given by

$$\mathrm{L}\ (\mathrm{x},\ \mathrm{y},\ \mathrm{k}\sigma)=G\ (x,y,k\sigma)*\ I(x,y) \qquad (3)$$

Hence a DoG image between scales Kiσ and Kjσ is just the difference of the Gaussian-blurred images at scales Kiσ and Kjσ. For scale space extrema detection in the SIFT algorithm, the image is first convolved with Gaussian-blurs at different scales. The convolved images are grouped by octave (an octave corresponds to doubling the value ofKjσ), and the value of Ki is selected so that we obtain a fixed number of convolved images per octave. Then the Difference-of-Gaussian images are taken from adjacent Gaussian-blurred images per octave.

3.5Keypoint Localization

After scale space extrema are detected (their location being shown in the uppermost image) the SIFT algorithm discards low contrast keypoints (remaining points are shown in the middle image) and then filters out those located on edges. Resulting set of keypoints is shown on last image.

Scale-space extrema detection produces too many keypoint candidates, some of which are unstable. The next step in the algorithm is to perform a detailed fit to the nearby data for accurate location, scale, and ratio of principal curvatures. This information allows points to be rejected that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge.

3.6Interpolation of nearby data for accurate position

For each candidate keypoint, interpolation of nearby data is used to accurately determine its position. The initial approach was to just locate each keypoint at the location and scale of the candidate keypoint. The new approach calculates the interpolated location of the extremum, which substantially improves matching and stability.

The interpolation is done using the quadratic Taylor expansion of the Difference-of-Gaussian scale-space function, D (x, y, σ) with the candidate keypoint as the origin. This Taylor expansion is given by

$$D(x) = D + \frac{\partial D^T}{\partial X}X + \frac{1}{2}X^T\frac{\partial^2 D}{\partial X^2}X \qquad (4)$$

where D and its derivatives are evaluated at the candidate keypoint and X (x, y, σ) is the offset from this point. The location of the extremum, $\hat{X}$, is determined by taking the derivative of this function with respect to X and setting it to zero. If the offset X is larger than 0.5 in any dimension, then that's an indication that the extremum lies closer to another candidate keypoint. In this case, the candidate keypoint is changed and the interpolation performed instead about that point. Otherwise the offset is added to its candidate keypoint to get the interpolated estimate for the location of the extremum.

3.7Discarding low-contrast keypoints

To discard the keypoints with low contrast, the value of the second-order Taylor expansion D(X) is computed at the offset $\hat{X}$. If this value is less than 0.03, the candidate keypoint is discarded. Otherwise it is kept, with final scale-space location Y + $\hat{X}$, where Yis the original location of the keypoint.

3.8Eliminating edge responses

The DoG function will have strong responses along edges, even if the candidate keypoint is not robust to small amounts of noise. Therefore, in order to increase stability, we need to eliminate the keypoints that have poorly determined locations but have high edge responses.

For poorly defined peaks in the DoG function, the principal curvature across the edge would be much larger than the principal curvature along it. Finding these principal curvatures amounts to solving for the eigenvalues of the second-order Hessian matrix, H.

The eigenvalues of H are proportional to the principal curvatures of D. It turns out that the ratio of the two eigenvalues, say α is the larger one, and β the smaller one, with ratio , is sufficient for SIFT's purposes. The trace of H, i.e., , gives us the sum of the two eigenvalues, while its determinant, i.e., , yields the product.

3.9Orientation assignment

In this step, each keypoint is assigned one or more orientations based on local image gradient directions. This is the key step in achieving invariance to rotation as the keypoint descriptor can be represented relative to this orientation and therefore achieves invariance to image rotation.

First, the Gaussian-smoothed image L (x, y, σ) at the keypoint's scale σ is taken so that all computations are performed in a scale-invariant manner. For an image sample L (x, y) at scale σ, the gradient magnitude, m (x, y), and orientation, $\sigma\ (x, y)$, are precomputed using pixel differences

$$m\ (x, y) = \sqrt{(L\ (x+1, y) - L\ (x-1, y))^2 + (L\ (x, y+1) - L\ (x, y-1))^2} \qquad (5)$$

$$\theta\ (x, y) = \text{atan2}\ (L\ (x, y+1) - L(x, y-1), L(x+1, y) - L(x-1, y)) \quad (6)$$

The magnitude and direction calculations for the gradient are done for every pixel in a neighboring region around the keypoint in the Gaussian-blurred image L. An orientation histogram with 36 bins is formed, with each bin covering 10 degrees. Each sample in the neighboring window added to a histogram bin is weighted by its gradient magnitude and by a Gaussian-weighted circular window with σ that is 1.5 times that of the scale of the keypoint. The peaks in this histogram correspond to dominant orientations. Once the histogram is filled, the orientations corresponding to the highest peak and local peaks that are within 80% of the highest peaks are assigned to the keypoint. In the case of multiple orientations being assigned, an additional keypoint is created having the same location and scale as the original keypoint for each additional orientation.

3.10Keypoint descriptor

First a set of orientation histograms is created on 4x4 pixel neighborhoods with 8 bins each. These histograms are computed from magnitude and orientation values of samples in a 16 x 16 region around the keypoint such that each histogram contains samples from a 4 x 4 sub region of the original neighborhood region. The magnitudes are further weighted by a Gaussian function with σ equal to one half the width of the descriptor window. The descriptor then becomes a vector of all the values of these histograms. Since there are 4 x 4 = 16 histograms each with 8 bins the vector has 128 elements. This vector is then normalized to unit length in order to enhance invariance to affine changes in illumination. To reduce the effects of non-linear illumination a threshold of 0.2 is applied and the vector is again normalized.

Although the dimension of the descriptor, i.e. 128, seems high, descriptors with lower dimension than this don't perform as well across the range of matching tasks and the computational cost remains low due to the approximate BBF (see below) method used for finding the nearest-neighbor. Longer descriptors continue to do better but not by much and there is an additional danger of increased sensitivity to distortion and occlusion. It is also shown that feature matching accuracy is above 50% for viewpoint changes of up to 50 degrees. Therefore SIFT descriptors are invariant to minor affine changes.

4.. Radon transform

It is the integral transform consisting of the integral of a function over straight lines.

The Radon transform is widely applicable to tomography, the creation of an image from the projection data associated with cross-sectional scans of an object.

The Radon transform data is often called a sinogram because the Radon transform of a Dirac delta function is a distribution supported on the graph of a sine wave. Consequently the Radon transform of a number of small objects appears graphically as a number of blurred sine waves with different amplitudes and phases.

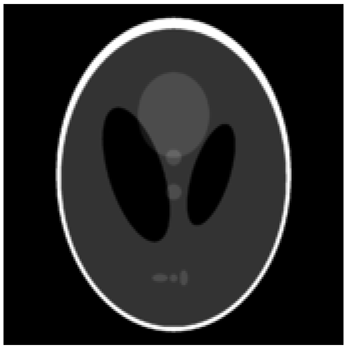


Fig 3 (a)

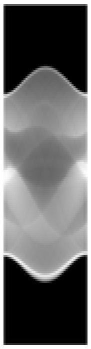


Fig 3 (b) Radon Transform

The Radon transform is useful in computed axial tomography (CAT scan), barcode scanners, electron microscopy of macromolecular assemblies like viruses and protein complexes, reflection seismology and in the solution of hyperbolic partial differential equations. It has the advantage of being more intuitive and have solid mathematical basis and used where accuracy is crucial.

5.Results And Discussion

Palmprint recognition system is implemented using MATLAB R2013a.

5.1. Input image

The Left and Right palmprint image of the person is captured as input image. The size of the input image is 256 x 256.

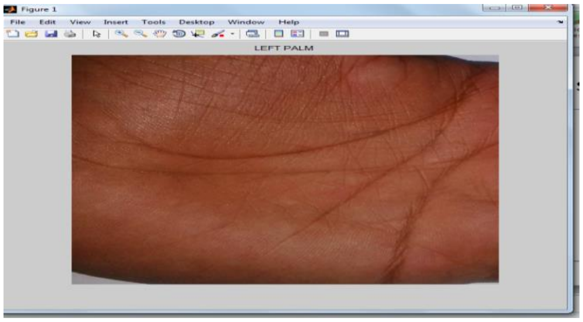


Fig 4 Left palmprint Image

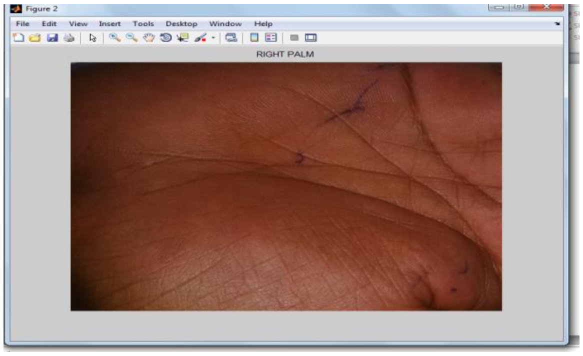


Fig 5 Right palmprint Image

Since we are considering reverse right palmprint image, the right palmprint image is reversed using suitable function.

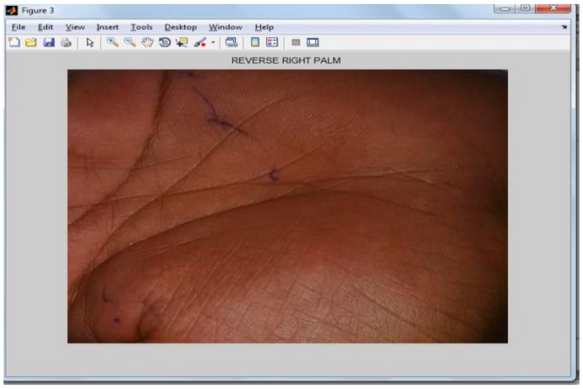


Fig 6 Reverse Right palmprint image

5.2.Fused image

The left and the reverse right images which was acquired as the input to the system is fused using standard fusion technique.

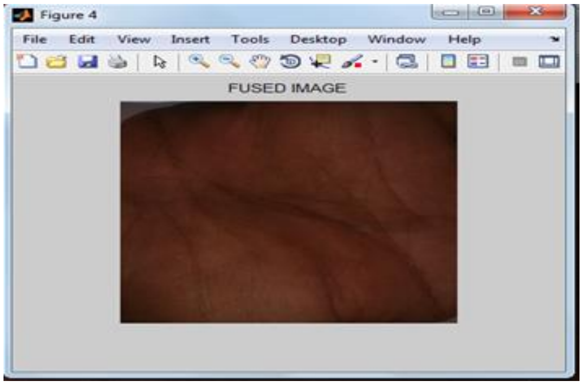


Fig 7 Fused Image

5.3. Preprocessed image

RGB image converted to Grayscale image and noise removed using median filter. Median filter is commonly used in preprocessing stage because it preserves the edges while removing the noise.

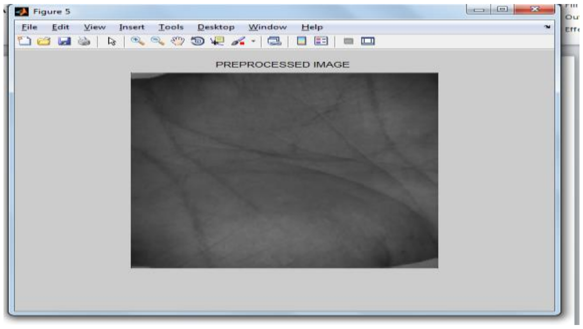


Fig 8 Preprocessed Image

5.4. Edge detection

Palm line detection is done using Radon transform which is commonly used in case of edge detection because it is more intuitive and used in the places where accuracy is more critical.

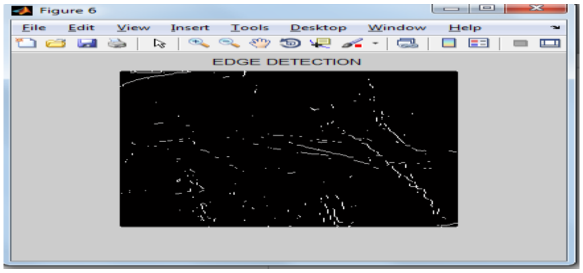


Fig 9 Image for Edge detection

5.5 Keypoints extraction using SIFT

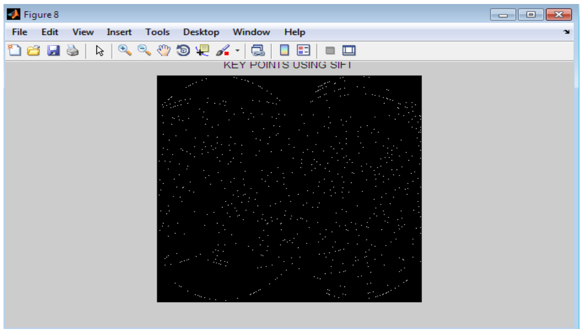
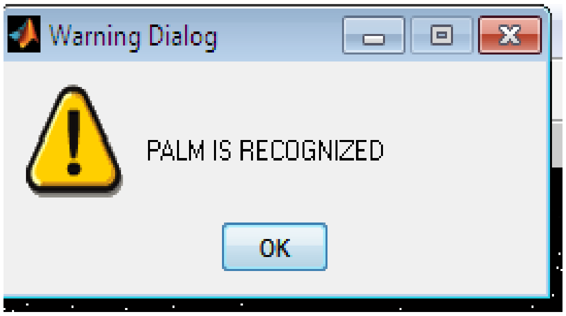


Fig 10Keypoints using SIFT

Keypoints are extracting using keypoint descriptor. More details about descriptor is discussed in section 5.6

5.6Palm recognition

If the features extracted from the acquired image matches with that of the database image the palm is recognized. For the input data given the result is palm recognized.



Warning Dialog

PALM IS RECOGNIZED

Fig 11 Image for Palm been Recognised

5.7 Comparison Tabulation

This section shows the comparison of different parameters between the proposed and existing systems.

The comparison between the performance parameters such as MSE, PSNR and computation time is given in the above table 4.1. Even though much variation in the parameters is not seen the security of the system is increased in the proposed system.

Table 1 SIFT Single palmprintvs SIFT Combined palmprint

Parameters	**SIFT for single palmprint image**	**SIFT for combined palmprint image**
MSE	1.7382	1.6556
PSNR	45.8456	45.9412
Computation Time	3.6445	3.6532

6. Conclusion and Future Scope

Multibiometrics involved in this by combining both left and right palm print gives better identification accuracy with high standard security. The proposed work shows how to utilize the shape of the palm to extract features using SIFT algorithm helped in increasing the performance and accuracy of the system. Further work has to done while fusing the left and right palm print by following suitable technique and algorithms can be adopted for palmprint recognition part to improve the performance of the system.

The experiments results showed that the accuracy is 98.7% and the response time is 3.65sec, which ensures that the method is very effective for personal authentication with high security.

References

[1] Dai.J, Zhou.J (2011) “Multifeature-based high-resolution palmprint recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no.5, pp.945–957.

[2] Du.F, Yu.P, Li.H and Zhu.L (2011) “Palmprint recognition using Gabor feature-based bidirectional 2DLDA,” Commun. Comput. Inf. Sci., vol. 159, no. 5, pp. 230–235.

[3] Guo.Z, Wu.G, Chen.QandLiu.W (2011) “Palmprint recognition by a two-phase test sample sparse representation,” in Proc. Int. Conf. Hand-Based Biometrics (ICHB), pp.1–4.

[4] Han.D, Guo.Z and Zhang.D (2008) “Multispectral palmprint recognition using wavelet-based image fusion,” in Proc. IEEE 9th Int. Conf. Signal Process, pp. 2074–2077.

[5] Huang D.S, Jia.W, Zhang.D (2008) "Palmprint verification based on principal lines," Pattern Recognit., vol. 41, no. 4, pp. 1316–1328.

[6] Imtiaz.H, Fattah S.A (2010) "DCT – based feature extraction algorithm for palm-print recognition", IEEE International Conference on Communication Control and Computing Technologies.

[7] Jain A.K, Ross.A, Prabhakar S. (2004) "An introduction to biometric recognition," IEEE Trans. Circuits Syst. Video Technol., vol. 14, no. 1, pp. 4–20.

[8] Kong A.W.K, Zhang.D (2004) "Competitive coding scheme for palm-print verification," in Proc. 17th Int. Conf. Pattern Recognit., vol. 1, pp.520–523.

[9] Kong A.W.K, Zhang.D and Kamel M. S (2009) "A survey of palmprint recognition," Pattern Recognit., vol. 42, no. 7, pp. 1408–1418.

[10] Kong.A, Zhang.D, Kamel.M (2006) "Palmprint identification using feature-level fusion," Pattern Recognit., vol. 39, no. 3, pp. 478–487.

[11] Ribaric.SandFratric. I (2005) "A biometric identification system based on eigenpalm and eigenfinger features," IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 11, pp. 1698–1709.

[12] Sun.Z, Tan.T, Wang.Y and Li S.Z (2005) "Ordinal palmprintrepresentaion for personal identification [represention read representation]," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., vol. 1, pp.279–284.

[13] Wu.X, Zhao.Q, Bu.W (2014) "A SIFT-based contactless palmprint verification approach using iterative RANSAC and local palmprint descriptors," Pattern Recognition, vol. 47, no. 10, pp. 3314–3326.

[14] Xu.Y, Zhang.D, Yang.J and Yang J. Y (2011) "A two-phase test sample sparse representation method for use with face recognition," IEEE Trans. Circuits Syst. Video Technol., vol. 21, no. 9, pp. 1255–1262.

[15] Zhang.D, Guo.Z, Lu.G, Zhang. D and Zuo.W (2010) "An online system of multispectral palmprint verification," IEEE Trans. Instrum. Meas.,vol. 59, no. 2, pp. 480–490.

[16] Zhang.D, Zuo.W and Yue.F (2012) "A comparative study of palmprint recognition algorithms," ACM Comput. Surv., vol. 44, no. 1, pp. 1–37.

[17] Zuo.W, Lin.Z, Guo.Z and Zhang.D (2010) "The multiscale competitive code via sparse representation for palmprint verification," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp.2265–2272.