Finger Vein Recognition Based on Multi-Orientation Weighted Symmetric Local Graph Structure

Song Dong[1], Jucheng Yang[1*], Yarui Chen[1], Chao Wang[1], Xiaoyuan Zhang[1] and Dong Sun Park[2]
[1] College of Computer Science and Information Engineering Tianjin University of Science and Technology, Tianjin, China.
[e-mail:jcyang@tust.edu.cn]
[2] Department of Electronic and Information Engineering, Chonbuk National University, Jeonbuk, Republic of Korea.
*Corresponding author: Jucheng Yang

Received August 7, 2014; revised October 12, 2014; revised December 8, 2014; accepted January 21, 2015; published October 31, 2015

Abstract

Finger vein recognition is a biometric technology using finger veins to authenticate a person, and due to its high degree of uniqueness, liveness, and safety, it is widely used. The traditional Symmetric Local Graph Structure (SLGS) method only considers the relationship between the image pixels as a dominating set, and uses the relevant theories to tap image features. In order to better extract finger vein features, taking into account location information and direction information between the pixels of the image, this paper presents a novel finger vein feature extraction method, Multi-Orientation Weighted Symmetric Local Graph Structure (MOW-SLGS), which assigns weight to each edge according to the positional relationship between the edge and the target pixel. In addition, we use the Extreme Learning Machine (ELM) classifier to train and classify the vein feature extracted by the MOW-SLGS method. Experiments show that the proposed method has better performance than traditional methods.

Keywords: Finger Vein Recognition, Multi-Orientation Weighted Symmetric Local Graph Structure, Symmetric Local Graph Structure, Local Graph Structure

This work was supported by the National Natural Science Foundation of China (Grant No. 61402332), Open Fund of Guangdong Provincial Key Laboratory of Petrochemical Equipment Fault Diagnosis No.GDUPTKLAB201334.

http://dx.doi.org/10.3837/tiis.2015.10.020 **ISSN : 1976-7277**

1. Introduction

Biometrics [1] [2], are identity authentication methods using physical characteristics, such as those of the face, fingerprints, irises, and veins, and so on. Compared with traditional features, such characteristics are safe, unique, lifelong invariant, and therefore widely used. Biometric systems typically include registration and authentication two stages.

In 1992, researchers in Hokkaido University first demonstrated that no vein characteristics are exactly the same for any two people in the world, even identical twins [3]. Finger vein authentication [4] [5] offers more security, stability, universality, liveness, and non-contact than a fingerprint test. It also has advantages over hand vein and palm vein tests: it requires a smaller volume; it has higher credibility and adequate sources of information [6]; and it offers useful redundancy: after the information entry registration for multiple fingers, if one finger is injured, we can still test the others. However, finger vein recognition is easily influenced by light condition [7], scattering [8], ambient temperature [9], and other factors.

For finger vein recognition, feature extraction is one of the key steps. Subspace methods are widely used. In 1991, Turk et al. [10] proposed using eigenfaces (also called Principal Component Analysis, PCA) for face recognition, and it is used for finger vein recognition [11] too. Linear Discriminant Analysis (LDA) [12], Two Dimensional Principal Component Analysis (2DPCA) [13], and Independent Component Analysis (ICA) [14] algorithms have also been proposed. In 1996, Ojala et al. [15], proposed a LBP algorithm and introduced the local information to feature extraction, which measured and extracted the local texture information of image with illumination invariance, and greatly improved the efficiency of facial recognition. However, the LBP operator was unable to extract large texture features. Ojala et al. [16] thus improved it by extending the 3×3 neighborhood to any neighborhood, and using the round neighborhood instead of the square neighborhood. The circular LBP operator only considers 8 neighborhood pixels, which makes use of the surrounding pixels deficient; therefore, in 2011, Rosdi et al. [17] extended the LBP operator in two directions, making it an LLBP operator, with more efficient feature extraction. They applied LLBP to finger vein recognition. However, LLBP only extracts horizontal and vertical line patterns. Therefore, Lu et al. [18] proposed the Generalized Local Line Binary Pattern (GLLBP) method, using a circular neighborhood of multi-direction feature extraction, and applied it to the vein recognition. In 2012, Meng et al. [19] proposed a Local Directional Code (LDC) feature extraction method to finger vein recognition by coding the image with the gradient information. Yang et al. [20] proposed a personalized best bit map (PBBM) algorithm and applied it to finger vein recognition. The PBBM is based on LBP, and in the matching phase, it only uses the best bits to improve recognition performance. In order to solve the common rotation problem of the finger vein images, Pang et al. [21] applied the SIFT feature extraction method, with good results. Xie et al. [22] proposed a Guided Gabor filter to further improve the extraction of finger vein features.

In 2011, inspired by the dominating set, Abusham et al. [23] first proposed the Local Graph Structure (LGS) algorithm and applied it to face recognition, which improved results resisted the influence of light. In 2014, Mohd et al. [24] proposed a symmetrical LGS (SLGS) algorithm. Compared with the LGS algorithm, the SLGS algorithm utilized the spatial information of the pixels in balance. However, SLGS has some deficiencies. When the SLGS operator assigns weight for each edge, it does not take the positional relationship between the edge and the target pixel into account, and the weight of the left pixels is too large, which does

not take full advantage of the spatial information of pixels around the target pixel. In addition, the SLGS operator only uses the zero-degree direction information of the target pixels, not the location and gradient information of the surrounding pixels in the other direction.

To overcome the shortcomings of traditional methods, this paper proposes a new method named Multi-Orientation Weighted Symmetric Local Graph Structure (MOW-SLGS), and uses the Extreme Learning Machine (ELM) to train and classify the feature. In the neighborhood of 5×5, we not only use the relationship between the target pixel and the surrounding pixels, but also the relationships between surrounding pixels. Moreover, we extract the features in 0 degree, 45°, 90°, and 135° direction by taking advantage of location relationship between pixels and gradient relationship and making full use of surrounding pixels. ELM [25] is a single hidden-layer feedforward neural network, fast and effective at classification. Experiments show the proposed method has better performance than traditional methods.

The second part of this paper introduces the theories of LGS and SLGS; the third part demonstrates the proposed method; the fourth part shows the experimental results, and the fifth part gives the conclusion.

2. Related Theory

2.1 LGS

The LGS method can weaken the influence of light on efficiency and improve calculation efficiency, which can be applied to the real-time systems. When assigning weights to the target pixel, it considers all relationships between pixels, not just those which directly include the target pixel, which improves recognition.

The LGS [26] algorithm is dominated by the dominating set in graph structure. Given an undirected graph G = <V, E>, where V is the point set whose size is n and E is the set of edges, then S, a subset of V, can be called a dominating set if and only if for any point v in V-S, there is a point u of S and $(u, v) \in E$. That is, for any vertex v in G, if v either belongs to D or is adjacent to a vertex in D, then D is called a dominating set of G.

For each target pixel, the algorithm selects 5 surrounding pixels and the composition of the surrounding structure is as shown in **Fig. 1**. Starting from the target pixel, at the left side of the target pixel, the algorithm compares the values of pixels one by one, counterclockwise, and if the pixel in the direction of the arrow's value is greater than the previous pixel value, then the connection between the two pixel sides is set to 1. Otherwise, it is set to 0. To the right of the target pixel, values are also compared one by one in a clockwise direction.

Through the above process, we can obtain 8 binary values, which are combined into a binary string to obtain the final value of the target pixel. As shown in **Fig. 2**, we calculate the feature value of the target pixel with LGS operator, like so:

$$\text{Feature}(00110100)=0\times128+0\times64+1\times32+1\times16+0\times8+1\times4+0\times2+0\times1=52$$

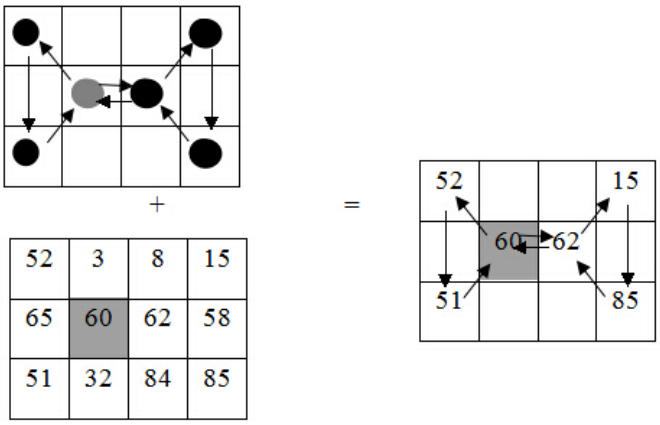


Fig. 1. LGS operator graph structure

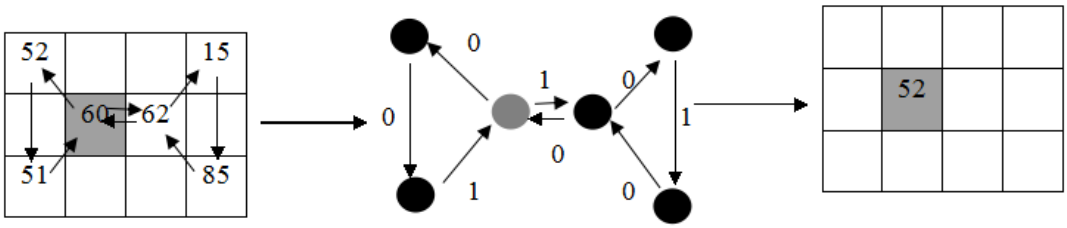


Fig. 2. Process of LGS operator calculations

2.2 SLGS

The SLGS algorithm [24] for texture-based image recognition has remarkably improved performance. It makes better use of spatial information than other feature extraction methods, and uses a more balanced relationship between surrounding pixels than LGS.

In **Fig. 2**, the target pixel used the two pixels on the left side and three pixels on the right side of the target pixel. This asymmetry is a problem with LGS. SLGS algorithm, as shown in **Fig. 3**, adds the idea of a symmetric algorithm into LGS, using symmetrical structure to express the relationship between adjacent pixels, making better use of the spatial information between the image pixels for the target pixel. It symmetrically selects 6 surrounding pixels, starting from the target pixel; at the left side of the target pixel, the algorithm compares the values of pixels one by one counterclockwise as before, and to the right, clockwise as before. As shown in **Fig. 4**, we calculate the feature value of target pixel with SLGS operator like so:

$$\text{Feature}(00101010)=0\times128+0\times64+1\times32+0\times16+1\times8+0\times4+1\times2+0\times1=42$$

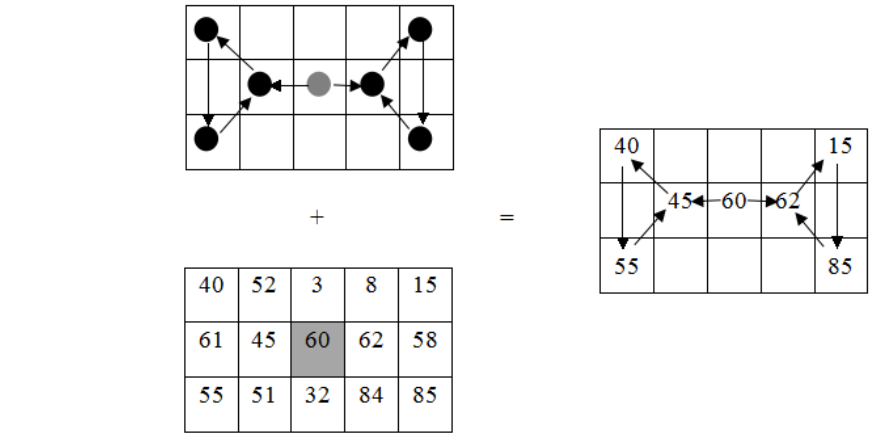


Fig. 3. SLGS operator graph structure

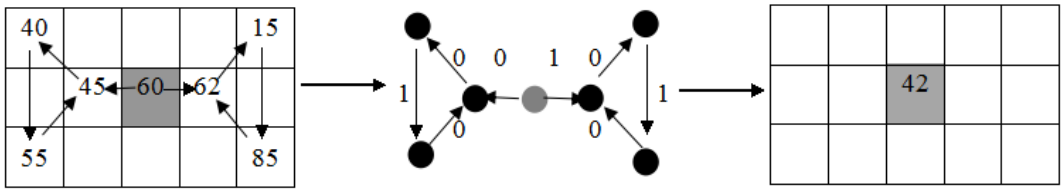


Fig. 4. Process of SLGS operator calculations

By comparing the magnitude between pixels adjacent to each other and giving different weights, we obtain a component of 8 binary strings and convert it to an integer, which is the target pixel.

3. Proposed Method

3.1 MOW-SLGS

When the SLGS operator assigns weight for each edge, it does not take into account the positional relationship between the edge and the target pixel, denying itself full advantage of the spatial information of pixels around the target pixel. The weight of the left pixels is too large. To balance the weight of right and left sides, this paper assigns weight to each edge according to the position relationship between the edge and the target pixel. We balance the weight of pixels on the right side and left side of the target pixel, and multiply the edge's weight by distance between surrounding pixel and target pixel. The Weighted SLGS (W-SLGS) operator is shown in **Fig. 5**.

In addition, because the SLGS operator only uses the 0° direction information of the target pixels, taking location information and direction information between the image pixels into account, this paper proposes an MOW-SLGS operator.

The proposed method is as shown below.

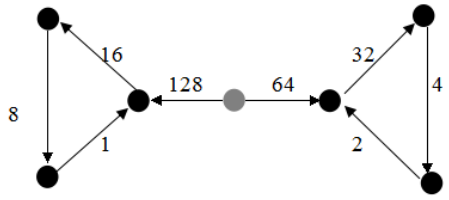


Fig. 5. The design of weights for Weighted SLGS

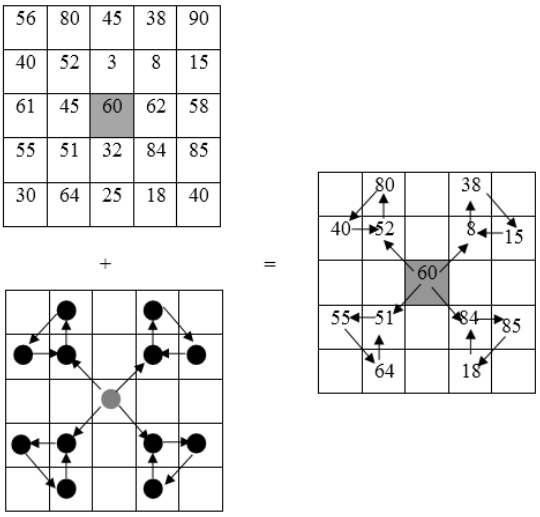


Fig. 6. MOW-SLGS operator graph structure in 45° and 135° direction

As shown in **Fig. 6** and **Fig. 7**, we do SLGS operations for the target pixel in the 0°, 45°, 90°, and 135° directions. In the 45° and 135° direction, as shown in **Fig. 8**, starting from the target pixel and going to the upper left or lower left, the algorithm compares the values of pixels one by one, counterclockwise, as before. The other two angles compare in a clockwise direction as before.

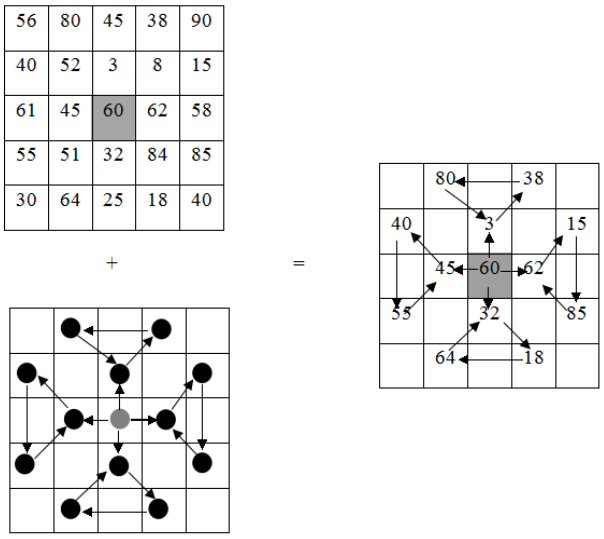


Fig. 7. MOW-SLGS operator graph structure in 0° and 90° direction

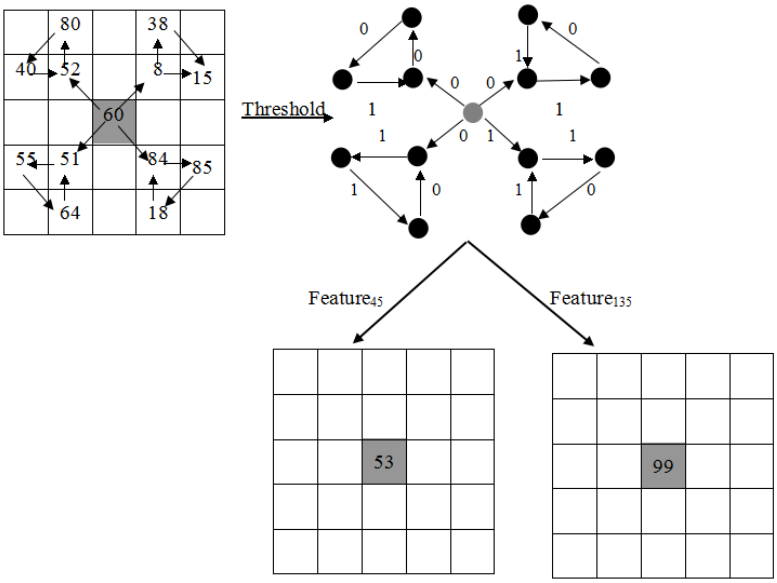


Fig. 8. Calculation process of MOW-SLGS operator in 45° and 135° direction

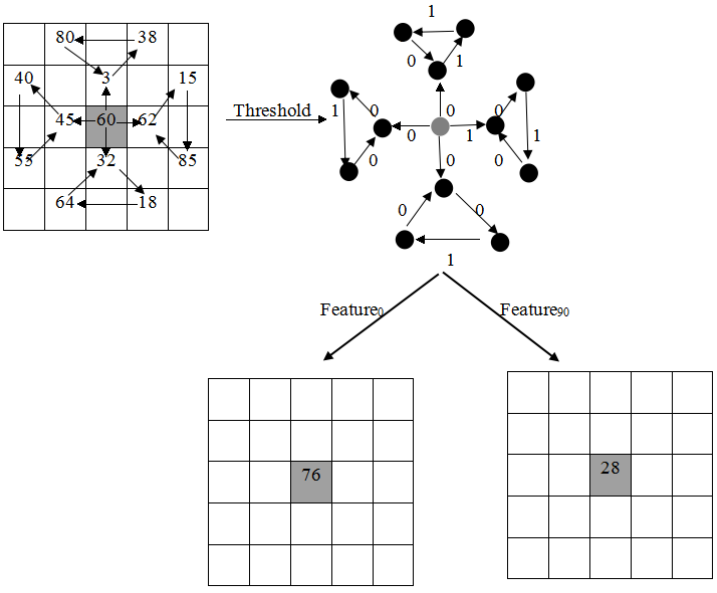


Fig. 9. Calculation process of MOW-SLGS operator in 0° and 90° direction

In 0°direction, the comparative method is the same as that of SLGS. In 90° direction, as shown in **Fig. 9**, starting from the target pixel, and going up, the algorithm compares the values of pixels counterclockwise as usual, and going down, clockwise as usual.

After processing with the MOW-SLGS operation, we can obtain 4 feature values: $Feature_0$, $Feature_{45}$, $Feature_{90}$, and $Feature_{135}$. The bigger the change in the direction feature value, the

bigger the change in pixels. Therefore, the direction can better reflect the position information and gradient information of pixels around the target pixel. So in this paper, we choose the maximum value as the feature value of the target pixel. As shown in **Fig. 8** and **Fig. 9**, we calculate the feature value of target pixel with MOW-SLGS operator as follows:

$Feature_0(01001100)=0\times128+1\times64+0\times32+0\times16+1\times8+1\times4+0\times2+0\times1=76$

$Feature_{45}(00110101)=0\times128+0\times64+1\times32+1\times16+0\times8+1\times4+0\times2+1\times1=53$

$Feature_{90}(00011100)=0\times128+0\times64+0\times32+1\times16+1\times8+1\times4+0\times2+0\times1=28$

$Feature_{135}(01100011)=0\times128+1\times64+1\times32+0\times16+0\times8+0\times4+1\times2+1\times1=99$

$Feature=\max\{Feature_0, Feature_{45}, Feature_{90}, Feature_{135}\}=Feature_{135}=99$

In the MOW-SLGS operator, we set the weight according to the distance between pixels, and the closer the distance, the greater the impact on the target pixel, so the weight we assign corresponds with the image distribution in space. And the MOW-SLGS operator makes full use of position and gradient information, making it more resistant to changes in light, temperature, and other external conditions. Additionally, we extend the operator in four directions, enriching the biometrics and improving feature extraction.

3.2 Dimensionality Reduction

To extract features better and remove redundancy and noise from finger vein features, we use the PCA algorithm [10] to reduce the dimensionality of the feature matrix. When the contribution rate is 0.95, the main feature extracted by PCA algorithm can already describe the overall characteristics, while the characteristic dimension is not large, and the recognition performance is not very different from the performance when contribution rate is greater than 0.95.In the experiment, we select 0.95 as the contribution rate.

3.3 Training and Classification

In our experiments, we use the ELM [26] for training and classification.

A standard single-hidden-layer feedforward neural network which contains N training samples $\{x_i, y_i\}$ (i = 1, ..., N), $\tilde{N}$ hidden-layer nodes and has excitation function g(x) can be expressed as follows:

$$\sum_{i=1}^{\tilde{N}} \beta_i g(\omega_i \bullet x_j + b_i) = o_j, \quad j=1, \quad 2, \quad \cdots, \quad N \tag{1}$$

Where, ω_i represents the input weights of the input neuron and the i-th hidden-layer node; β_i is the output weights of i-th hidden-layer nodes and output neurons; b_i is the bias of i-th hidden-layer node; o_j represents the output value of j-th sample.

The process of ELM algorithm is as follows:

Step 1: Set the input random weights ω_i, and bias b_i, i = 1, ..., $\tilde{N}$;

Step 2: Calculate the hidden-layer output matrix H;

Step 3: Calculate the output weights β: $\hat{\beta}=H^{+}Y$.

Compared with other training algorithm, ELM algorithm does not need to adjust the input weights and the bias of hidden network neurons, and we can get a unique optimal solution at higher efficiency.

In our experiments, we set the number of hidden neurons to 2000, the reason will be given in Section 4.

4. Experiment and analysis

4.1 Experimental database

The database used here is the open Homologous Multi-modal Traits Database [28] developed by Shandong University. During finger vein image acquisition, each person must provide the index finger, middle finger, ring finger of both hands. Each finger produces 6 images, meaning 36 images in total for one person. In this database, there are 106 persons, each with 36 images, for a total of 3816 images. The size of each image is 320 × 240 as shown in **Fig. 10**.

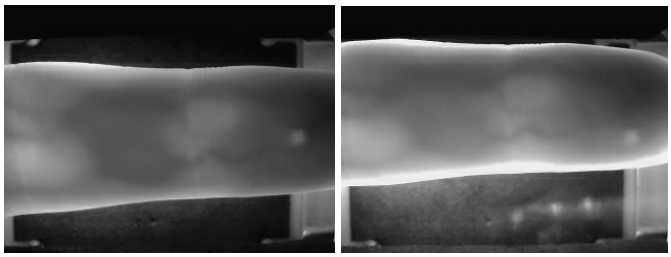


Fig. 10. Image of finger vein database

In our experiments, we select 100 different fingers from the original database, and each finger has 6 images, so we have a total of 600 images in the new database. We first crop the ROI images(**Fig. 11**), then use the Guided Gabor enhancement method on them [22], which yields image in the size of 256×96 (**Fig. 12**).

All the experiments are implemented with MATLAB 7.0, and performed on a PC with a 2.8 GHz CPU and 2.0 G memory in windows 7 OS.

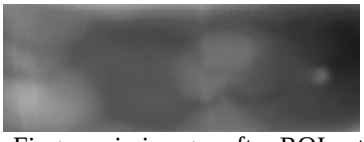


Fig. 11. Finger vein images after ROI extraction

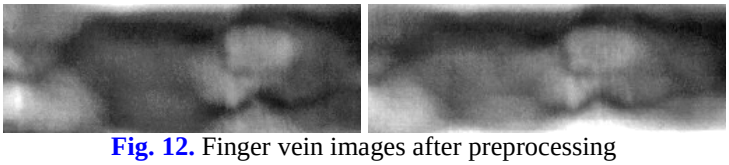


Fig. 12. Finger vein images after preprocessing

4.2 The experimental results

4.2.1 Determine the number of hidden neurons

We conducted the experiment to test the influence of hidden neurons on the performance of finger vein recognition system, and the result is shown in **Table 1**. For each finger, we used four images for training, two for testing. From the following table, we can see when the

number of hidden neurons is more than 1500, the recognition performance tends to be stable at about 0.9100. So, in order to save time, we choose 2000 as the number of hidden neurons.

Table 1. Recognition rate of different number of hidden neurons

the Number of Hidden Neurons	1000	1500	2000	2500	3000	4000	5000
Recognition Rate	0.8975	0.9100	0.9125	0.9115	0.9090	0.9080	0.9090

4.2.2 Determine the best image blocking method

As the size of the feature matrix depends on the number of block images processed, we investigate the effect of image blocking method on the experimental results. We first explore the best image blocking method. The image blocking method is the method to block a finger image to obtain the sub-images. The blocking method M × N means that we divide the image into M rows and N columns.

Since the number of sub-images determines the size of the image feature matrix, but when the number of sub-images is too large, the feature matrix is too big, it will cost a lot of time. Therefore, in the trial, we selected the following blocking methods from the original image: 1×1, 2×1, 2×2, 4×2, and 4×4. The experimental results are shown below. For each person in the above database, we used three images for training, the remaining three for testing, and the MOW-SLGS algorithm for testing.

Fig. 13 and **Table 2** show that, among different blocking methods, the recognition rates fluctuate slightly, but 4×2 produces the best rate; therefore, we use the 4×2 blocking method. Rates for 1×1, 2×1, 2×2, 4×2, and 4×4 blocks are 86.67%, 86.50%, 86.33%, 87.17%, and 87.00%, respectively.

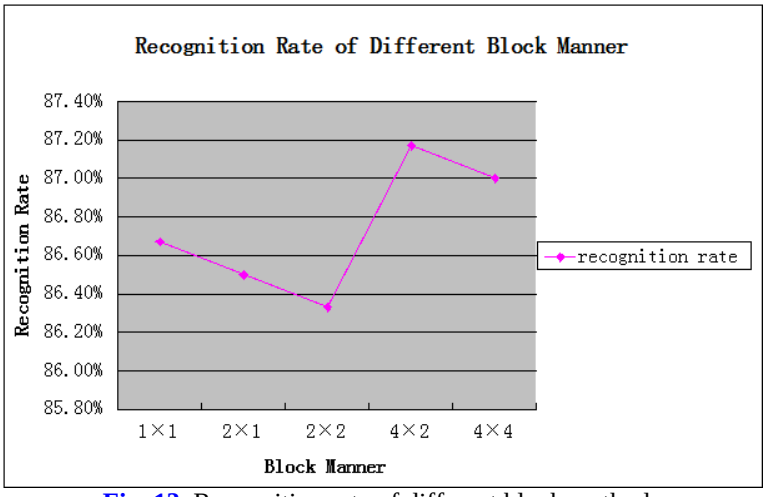


Fig. 13. Recognition rate of different block methods

Table 2. Recognition rates of different blocking method

Blocking method	1×1	2×1	2×2	4×2	4×4
Recognition rate	0.8667	0.8650	0.8633	0.8717	0.8700

4.2.3 Comparison of performance using different classifiers

In our experiments, we tested the performances by using three state-of-the-art classifiers: the Nearest Neighbor (1-NN), ELM, and VBELM [28]. Again we used four images for training, two for testing, and the MOW-SLGS algorithm for testing. In the training stage, we also use MOW-SLGS for feature extraction. Each experiment was repeated with 20 times and chose the average value as result. The results are shown as **Table 3**.

Table 3. Recognition rates of different classifiers

Classifier	Training time(s)	Testing time(s)	Recognition rate
1-NN	/	26.4529	0.8500
ELM	1.8188	0.0938	0.9125
VBELM	111.8531	0.1008	0.9105

From **Table 3**, we can see that the 1-NN method does not train the samples, so we just compare the training time between ELM and VBELM. When the number of training samples is 4, the training times for ELM and VBELM are 1.8188 s and 111.8531 s and the testing time for 1-NN, ELM, and VBELM are 26.4529 s, 0.0938 s, and 0.1008 s, respectively. While the recognition rate of 1-NN, ELM, and VBELM are 85.00%, 90.85%, and 91.25%, respectively. The training time of VBELM is almost 60 times that of ELM, and the recognition rate is lower than ELM. In addition, the recognition rate of ELM is higher than 1-NN, and its test time is lower than 1-NN's; therefore, this paper chose ELM as the classifier for training and classification.

4.2.4 Comparison of the processing time by using different algorithms

In our experiment, we compared the processing time of LBP, LGS, SLGS and MOW-SLGS, respectively, and they are shown in **Table 4**.

Table 4. The processing time of different algorithms

Algorithms	LBP	LGS	SLGS	MOW-SLGS
processing time（second）	0.4764	0.4764	0.4969	1.5517

From **Table 4**, we can see that the processing time of LBP, LGS, SLGS and MOW-SLGS is 0.4764 s, 0.4764 s, 0.4969 s, 1.5517 s, respectively. The processing time for MOW-SLGS is highest, because it uses four directions. MOW-SLGS's maximum processing time is almost four times that of SLGS.

4.2.5 Comparison of the performance of using the different algorithms

(I)Comparison between the weight and non-weight SLGS

In order to test the algorithm's performance by modifying the weights and increasing multi-direction, we compared the recognition rate of SLGS, W-SLGS and MOW-SLGS with the same database.

The experimental results are shown in **Table 5**. When the number of training samples is 1, 4, or 5, W-SLGS improves somewhat on SLGS; however, when the number of samples is 2 or 3, W-SLGS does not do as well as SLGS.

When the number of samples is 2, 3, or 5, MOW-SLGS outperforms W-SLGS, but when

the number of samples is 1 or 4, it does not even do as well as SLGS. However, overall, modifying weights and increasing multi-direction both improve the performance of SLGS algorithm.

Table 5. Recognition rate comparison between the weight and non-weight SLGS

Algorithms / Number of training samples (N)	SLGS	W-SLGS	MOW-SLGS
1	0.7930	0.8188	0.8130
2	0.8400	0.8210	0.8462
3	0.8517	0.8317	0.8717
4	0.9100	0.9205	0.9125
5	0.9500	0.9555	0.9600

(II) Comparison of the performance of different algorithms

This experiment compared the recognition rate of the LGS, SLGS, LBP, and MOW-SLGS methods under the same finger vein database, and for each set of experiments, we use the average of three repeated experiments as a result. The results are shown in Table 6. From the table, we can see that when the number of training samples (N) is 1, the recognition rate of the LGS, SLGS, LBP, and MOW-SLGS methods are 67.10%, 79.30%, 69.60%, and 81.30%, respectively. Of the algorithms, MOW-SLGS has the highest recognition rate. When N is 3, the recognition rate of the LGS, SLGS, LBP, and MOW-SLGS methods are 79.17%, 85.17%, 78.33%, and 87.17%, respectively, making MOW-SLGS again highest. When N is 5, the recognition rate of the LGS, SLGS, LBP, and MOW-SLGS methods are 94.50%, 95.00%, 95.00%, 96.00%, respectively, making all rates high, but MOW-SLGS still highest. MOW-SLGS thus has the best performance overall.

Since the finger vein image is rich in direction information, and LGS and SLGS only use the pixel information in the 0° direction of the target pixel, not that in the 45°, 90°, and 135° directions, they are less effective. The LBP operator uses only eight pixels around the target pixel and the magnitude relationship between the target pixel and the surrounding pixels, but ignores the gradient information between the surrounding pixels, costing it some feature extraction.

Table 6. Recognition rate of different algorithms under the same finger vein database

Algorithms / Number of training samples (N)	LGS	SLGS	LBP	MOW-SLGS
1	0.6710	0.7930	0.6960	0.8130
2	0.7350	0.8400	0.7538	0.8462
3	0.7917	0.8517	0.7833	0.8717
4	0.8825	0.9100	0.8825	0.9125
5	0.9450	0.9500	0.9500	0.9600

Fig. 14 shows the recognition rates of LGS, SLGS, PCA, LBP and MOW-SLGS more intuitively. As can be seen from it, the curve of MOW-SLGS is above other curves, and highest when the number of training samples is 5.

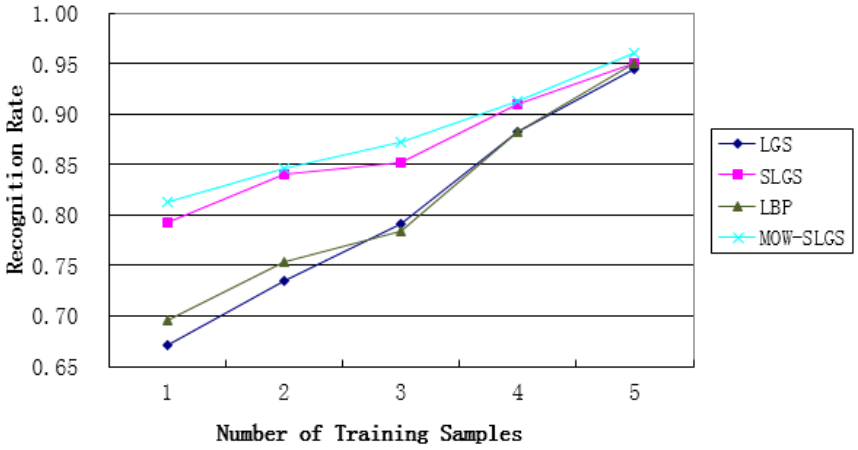


Fig. 14. Recognition rate of different algorithms under the same finger vein database

From another view, we draw the ROC curves [4] for LBP, LGS, SLGS, and MOW-SLGS, shown in **Fig. 15** under the same finger vein database with the 1-NN classifier. Again, the curve of MOW-SLGS is on bottom of the other curves, which means that MOW-SLGS shows the best performance. The false acceptance rate (FAR) and false rejection rate (FRR) are evaluated with 2×(100× (100-1)×4) =79200 imposter matches versus 2×(100×4)=800 genuine matches respectively. In the experiment, we use Mahalanobis distance as the matching method.

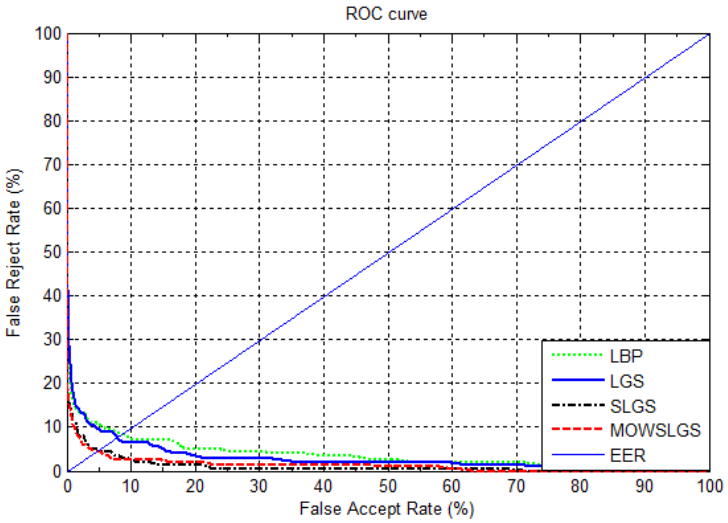


Fig. 15. ROC curves of the different algorithms

Table 7 compares EER rate of different algorithms under the same finger vein database with the 1-NN classifier. As we can see from the table, the EER rate of LBP, LGS, SLGS, MOW-SLGS are 8.52%, 7.82%, 5.00%, 4.50% respectively, which means MOW-SLGS show

better results compared with other algorithms.

Table 7. EER rate of different algorithms under the same finger vein database

Algorithms	LBP	LGS	SLGS	MOW-SLGS
EER rate (%)	8.52	7.82	5.00	4.50

5. Conclusion

This paper proposes a novel feature extraction (MOW-SLGS) method, and applies it to finger vein recognition. The MOW-SLGS method uses gradients and location of the pixels around target pixel, assigns weight to each edge according to the position relationship between the edge and the target pixel, and improves on SLGS in its resistance to changes in light and background factors. We test it with different classifiers (1-NN, ELM, VBELM). Experimental results show that the proposed method recognizes finger veins more easily than any other methods.

References

[1] A.K. Jain, P. Flynn and A.A. Ross, *Handbook of Biometrics*, 1st Edition, Berlin, 2008. Article (CrossRef Link)

[2] A.A. Ross, K. Nandakumar and A.K. Jain, *Handbook of Multibiometrics*, 1st Edition, Berlin, 2006. Article (CrossRef Link)

[3] M. Kono, H. Ueki and S. Umemura, "A new method for the identification of individuals by using of vein pattern matching of a finger," in *Proc. of Fifth Symposium on Pattern Measurement*, pp. 9-12, 2000.

[4] Y. Lu, S.J. Xie, S. Yoon, J.C. Yang and D.S. Park, "Robust Finger Vein ROI Localization Based on Flexible Segmentation," *Sensors*, vol. 13, pp.14339-14366, October, 2013. Article (CrossRef Link)

[5] Y. Lu, S. Yoon, S.J. Xie, J.C. Yang, Z.H. Wang and D.S. Park, "Efficient descriptor of histogram of salient edge orientation map for finger vein recognition," *Applied Optics*, vol. 53, issue 20, pp.4585-4593, July, 2014. Article (CrossRef Link)

[6] W. S. Song, T. J. Kim and H. C. Kim, "A finger-vein verification system using mean curvature," *Pattern Recogn Lett.*, vol. 32, issue 11, pp.1541–1547, August, 2011. Article (CrossRef Link)

[7] M. Kono, H. Ueki and S. Umemura, "Near-Infrared Finger Vein Patterns for Personal Identification," *Applied Optics*, vol. 41, issue 35, pp. 7429-7436, December, 2002. Article (CrossRef Link)

[8] H. Kim, E.J. Lee, G.J. Yoon, S.D. Yang, E.C. Lee and S.M. Yoon, "Illumination normalization for SIFT based finger vein authentication," *Lecture Notes in Computer Science*, vol. 7432, pp. 21-30, 2012. Article (CrossRef Link)

[9] J.F. Yang, B. Zhang and Y.H. Shi, "Scatting removal for finger-vein image restoration," *Sensors*, vol. 12, no. 3 pp.3627-3640, March, 2012. Article (CrossRef Link)

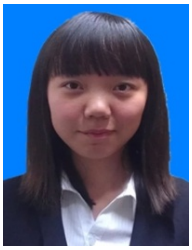
[10] M.A. Turk, A. P. Pentland, "Face Recognition Using Eigen Faces," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp.586-591, June 3-6, 1991. Article (CrossRef Link)

[11] J.D. Wu and C.T. Liu, "Finger-vein pattern identification using principal component analysis and the neural network technique," *Expert Systems with Application*, vol. 38, issue 5, pp.5423-5427, May, 2011. Article (CrossRef Link)

[12] A.M. Martinez and A.C. Kark, "PCA versus LDA," *Pattern Analysis and Machine Intelligence*, vol. 23, issue 2, pp.228-233,February, 2011. Article (CrossRef Link)

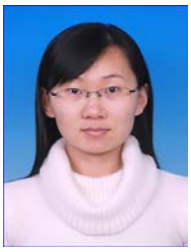
[13] G.P. Yang, X.M. Xi and Y.L. Yin, "Finger vein recognition based on (2D) 2PCA and metric

learning," *Biomedicine and Biotechnology*, pp.1–9, March, 2012. Article (CrossRef Link)
[14] C.J. Liu and J. Yang, "ICA Color Space for Pattern Recognition," *Neural Network*, vol. 20, no. 2, pp.248-257, October, 2009. Article (CrossRef Link)
[15] T. Ojala, M. Pietikainen and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition*, vol.29, issue 1, pp.51-59, January, 1996. Article (CrossRef Link)
[16] T. Ojala, M. Pietikainen and T. Maenpaa, "Multi-resolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Transactions on Pattren Anal.*, vol. 24, issue 7, pp. 971-987, July, 2002. Article (CrossRef Link)
[17] B.A. Rosdi, C.W. Shing and S.A. Suandi. "Finger vein recognition using local line binary pattern," *Sensors*, vol.11, pp.11357–11371, November, 2011. Article (CrossRef Link)
[18] Y. Lu, S. Yoon, J.C. Yang, S.J. Xie, Z.H. Wang and D.S. Park. "Finger Vein Recognition Using Generalized Local Line Binary Pattern," *KSII Transactions on Internet and Information Systems*, vol.8, no.5, pp.1766-1784, May, 2014.
[19] X.J. Meng, G.P. Yang, Y.L.Yin and R.Y. Xiao, "Finger vein recognition based on local directional code," *Sensors*, vol.12, pp.14937-14952, November, 2012. Article (CrossRef Link)
[20] G.P. Yang, X.M. Xi and Y.L. Yin, "Finger vein recognition based on a personalized best bit map," *Sensors*, vol.12, pp.1738-1757, February, 2012. Article (CrossRef Link)
[21] S.H. Pang, Y.L. Yin, G.P. Yang and Y.N. Li, "Rotation Invariant Finger Vein Recognition," in *Proc. of 7th Chinese Conference CCBR*, pp.151-156,December 4-5 2012. Article (CrossRef Link)
[22] S.J. Xie, J.C. Yang, S. Yoon, Y. Lu and D.S. Park, "Guided Gabor filter for finger vein pattern extraction," in *Proc. of the 8th International Conference on Signal Image Technology and Internet Based Systems*, pp.118-123, November 2012. Article (CrossRef Link)
[23] E. E. A. Abusham and H. K.Bashir, "Face recognition using local graph structure (LGS)," *Lecture Notes in Computer Science*, vol. 6762, pp.169–175, 2012. Article (CrossRef Link)
[24] F.A. Mohd, M.S. Sayeed, K. S. Muthu, H.K. Bashier, A. Afizan and S.Z. Ibrahim, "Face recognition with Symmetric Local Graph Structure (SLGS)," *Expert Systems with Applications*, vol.41, pp. 6131–6137, October, 2014. Artical(CrossRefLink)
[25] G. B. Huang, Q. Y. Zhu and C. K. Siew, "Extreme Learning Machine: Theory and Applications," *Neurocomputing*, vol. 70, no. 1-3, pp. 489-501, December, 2006. Artical(CrossRefLink)
[26] H.K. Bashier, E. A. Abusham, and F. Khalid, "Face Detection Based on Graph Structure and Neural Networks," *Trends in Applied Sciences Research*, vol.7,issue 8, pp. 683-691,2012. Article (CrossRef Link)
[27] S.J. Xie, J.C. Yang, S. Yoon, and D.S. Park, "Component-based Extreme Learning Machines for Finger Vein Recognition," *Cognitive Computation*, vol. 6, issue 3, pp. 446-461, September, 2014. Artical(CrossRefLink)
[28] http://mla.sdu.edu.cn/sdumla-hmt.html.
[29] Y.R. Chen, J.C. Yang, C. Wang and D.S. Park, "Variational Bayesian Extreme Learning Machine," *Neural Computing and Applications*, pp.1-12, September, 2014. Artical(CrossRefLink)



Song Dong is a master student in the College of Computer Science and Information Engineering, Tianjin University of Science and Technology, P.R. China. She received her B.S. degree from Tianjin University of Science and Technology, China, 2013. She has published several papers about biometrics. Her research interests include image processing, biometrics, pattern recognition, and neural networks. E-mail: dongsongmj@mail.tust.cn.

Jucheng Yang is a full professor in the College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, P.R.China. He is a Specially-appointed Professor of Tianjin City and Haihe Scholar. He received his B.S. degree from South-Central University for Nationalities, China in 2002, MS and PhD degrees from Chonbuk National University, Republic of Korea in 2004 and 2008. He did his postdoctoral work at the Advanced Graduate Education Center of Jeonbuk for Electronics and Information Technology-BK21 (AGECJEIT-BK21) in Chonbuk National University, too. He has published over 90 papers in related international journals and conferences, such as IEEE Trans. on HMS, IEEE Systems Journal, Expert Systems with Applications and so on. He has served as editor of five books in biometrics, and as reviewer or editor for international journals such as IEEE Transactions on Information Forensics & Security, IEEE Trans. on Circuits and Systems for Video Technology, IEEE Communications Magazine, and as the guest editor of Journal of Network and Computer Applications. He is the general chiar of CCBR'15, ISITC2015, and the publicity chair of ICMeCG'10-12. And he is the program committee member of many conferences such as JCeSBI'10, IMPRESS'11 and CCBR'13, CCBR'14. He owns 7 patents in biometrics. His research interests include image processing, biometrics, pattern recognition, and neural networks. E-mail: jcyang@tust.edu.cn.



Yarui Chen, PhD, is an associate professor in the College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, P.R.China.. She has published several papers about biometrics in conferences and journals. Her main research interests include artificial intelligence and machine learning. E-mail: yrchen@tust.edu.cn.

Chao Wang is a master student in the College of Computer Science and Information Engineering, Tianjin University of Science and Technology, P.R. China. He received his B.S. degree from Tianjin University of Science and Technology, China, 2012. His research interests are content-based multimedia processing, pattern recognition E-mail: 12834002@mail.tust.cn.

Xiaoyuan Zhang, is a master student in the College of Computer Science and Information Engineering, Tianjin University of Science and Technology, P.R. China. He received his B.S. degree from Tianjin University of Science and Technology, China, 2013. His reasearch interests include pattern recognition and image processing. E-mail: xyzhang@mail.tust.cn.

DongSun Park is a professor at the Chonbuk National University, Republic of Korea, and the head of the Advanced Graduate Education Center of Jeonbuk for Electronics and Information Technology -BK21. He received his BS from Korea University, Republic of Korea in 1979, and MS and PhD degrees from the University of Missouri, United States in 1984 and 1990. He has published many papers in international conferences and journals. He is a member of IEEE Computer Society. His research interests include image processing, image coding, pattern recognition, and neural networks. E-mail: dspark@jbnu.ac.kr.