

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Face Recognition with Facial Occlusion Based on Local Cycle Graph Structure Operator

Jucheng Yang, Lingchao Zhang, Meng Li,
Tingting Zhao, Yarui Chen, Jianzheng Liu and Na Liu

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.78597>

Abstract

Facial occlusion is a difficulty in the field of face recognition. The lack of features caused by occlusion may reduce the face recognition rate greatly. How to extract the identified features from the occluded faces has a profound effect on face recognition. This chapter presents a Local Cycle Graph Structure (LCGS) operator, which makes full use of the information of the pixels around the target pixel with its neighborhood of 3×3 . Thus, the recognition with the extracted features is more efficient. We apply the extreme learning machine (ELM) classifier to train and test the features extracted by LCGS algorithm. In the experiment, we use the olivetti research laboratory (ORL) database to simulate occlusion randomly and use the AR database for physical occlusion. Physical coverings include scarves and sunglasses. Experimental results demonstrate that our algorithm yields a state-of-the-art performance.

Keywords: face recognition, facial occlusion, local cycle graph structure

1. Introduction

With the development of science and technology, face recognition technology gradually appeared in our daily life. As the unique biometric key [1], people can use it as a password to protect personal property and information security; the police can arrest criminals through scanning faces based on the photos obtained from cameras [2]. All these applications require face recognition technology. However, most of the faces collected in those practical applications are occluded in some dynamic environments, such as scarves, sunglasses, and so on.

Thus, extracting the effective facial feature information from the occluded faces is the key issue for face recognition.

Many feature extraction algorithms have been proposed to extract facial features; the extracted features can be divided into global features [3] and local features [4–6]. It has been shown that the extracted global facial features cannot effectively solve the recognition problems with occluded faces [7]. On the other hand, local features could deal with face recognition with occluded faces fairly well.

Abusham et al. [8] proposed the Local Graph Structure (LGS) operator, which combines 5 pixels around the center pixel into a graph structure in the neighborhood of 3×4 . The LGS algorithm improves the recognition rate through the efficient use of pixel information in the neighborhood. However, the graph structure constructed by the LGS algorithm is unbalanced. Abdullah et al. [9] proposed a Symmetric Local Graph Structure (SLGS) algorithm to solve the unbalanced problem. However, SLGS only considers the pixel information on the left and right sides of the target pixel without analyzing the information in other directions. Thus, the information in the extracted features is still insufficient to achieve a good recognition rate. For this issue, Dong et al. [10] proposed MOW-SLGS algorithm, which calculates the characteristic value of pixels around the target pixel in the 5×5 neighborhood in the direction of 0, 45, 90, and 135°, respectively, and gives the optimal weight. Finally, the maximum value of the four directions is set as the eigenvalue. MOW-SLGS was shown to provide a reasonable recognition rate.

However, LGS and MOW-SLGS only choose several pixels in the neighborhood of 3×4 and 5×5 . When computing the feature values, they do not consider all the pixels. This will lead to some information loss. To solve this problem, this chapter proposes a Local Cycle Graph Structure (LCGS) operator, which constructs the graph structure in the neighborhood of 3×3 . The feature values of the target pixel are obtained by using all the pixels in its selected neighborhood. Due to the dimension of the matrix of the characteristic values is too large, the training for classification is not easy. Therefore, we employ the principal component analysis (PCA) [11, 12] method for dimensionality reduction through the experiments. We use the extreme learning machine (ELM) [13, 14] to classify and match the features after dimensionality reduction. Experimental results show that the LCGS algorithm performs better for face recognition with occlusion.

The rest of this article is organized as follows. Section 2 introduces the technical details of LGS and MOW-SLGS; Section 3 gives the detailed algorithm of LCGS; Section 4 shows the experimental results on the ORL and AR database; Section 5 concludes this manuscript.

2. Related theory

2.1. Local graph structure (LGS)

LGS algorithm [8] applied the graph structure in the calculation of the feature values, the main idea is as follows: take the center pixel as the target pixel, and take the two pixels of the left side and three pixels of the right side in the neighborhood of 3×4 to constitute the graph

structure. Then compare the pixel value of the target pixel and its two pixels on the left by the counterclockwise. Set to 1 when the pixel value becomes larger, to 0 when it becomes smaller. Three pixels on the right side of the target pixel are compared in the clockwise direction. The same as above, set to 1 when the pixel value becomes larger, to 0 when it becomes smaller. Finally, the target pixel value is compared with the adjacent pixel value on the right. The whole procedure is shown in **Figure 1**, where X_0 is the target pixel. The comparison order is conducted as $X_0 \rightarrow X_1 \rightarrow X_2 \rightarrow X_0 \rightarrow X_3 \rightarrow X_4 \rightarrow X_5 \rightarrow X_3 \rightarrow X_0$. The final result is composed by these eight binary numbers. Finally, we convert the 8-bit binary numbers to its decimal format.

A specific example of LGS operator is shown in **Figure 1**. By comparing the value of the pixel with its adjacent pixel along the direction of arrows, the binary value of X_0 is 10,001,011. As a result, the decimal value of the target pixel is obtained as 139.

2.2. Multi-orientation weighted symmetric local graph structure (MOW-SLGS)

MOW-SLGS algorithm [10] is improved based on the LGS algorithm, which mitigates the problems of the LGS algorithm by constructing the graph structure in the horizontal direction. The extracted feature for the target pixel is not sufficient, and the weight ratio is not balanced. The algorithm is implemented as follows: in the 5×5 neighborhood, we take the center pixel as the target pixel, and then the graph structure is constructed by the target pixel in 0, 45, 90, and 135° direction. Finally, we calculate the feature values of each direction. For the directions of 0, 45, and 135°, the left side of the target pixel is used to compare the value of the pixel in the counterclockwise direction, the right side of the target pixel to compare the value of the pixel in the clockwise direction. If the pixel value is larger, we set it to 1; otherwise set 0. For the direction of 90°, the comparison of the upper side of the target pixel with counter-clockwise order, the lower side of the target pixel with the counter-clockwise order. The binary value is read from the central pixel along each direction. Each direction is firstly in the counter-clockwise order, and then in the clockwise order. The weight of the 8-bit binary number obtained by the algorithm of MOW-SLGS in each direction is shown in **Figure 2**, where X_0 is the target pixel. Finally, the maximum value is obtained as the characteristic value.

We take the direction of 45° as the example as shown in **Figure 3**. It can be seen that the characteristic value of the target pixel in the direction of 45° is 01101101. Thus, the final value is 122 after calculating its corresponding decimal value.

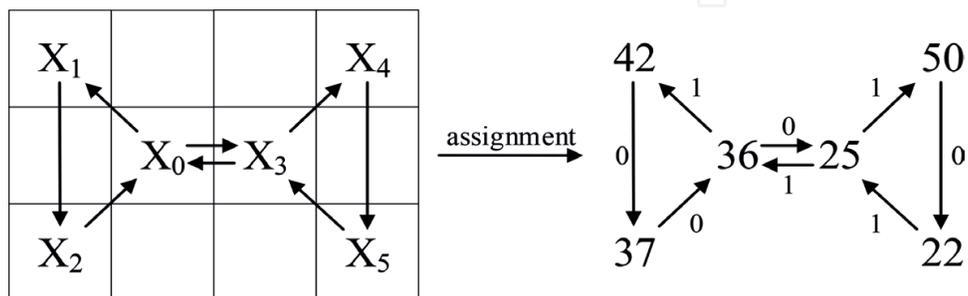


Figure 1. Template of LGS operator and its example.

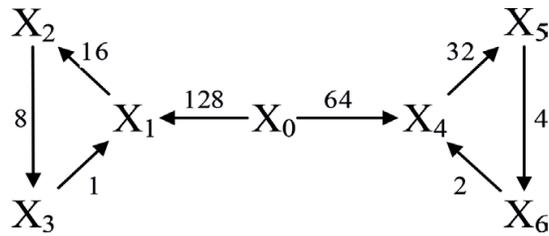


Figure 2. The design of weights for MOW-SLGS.

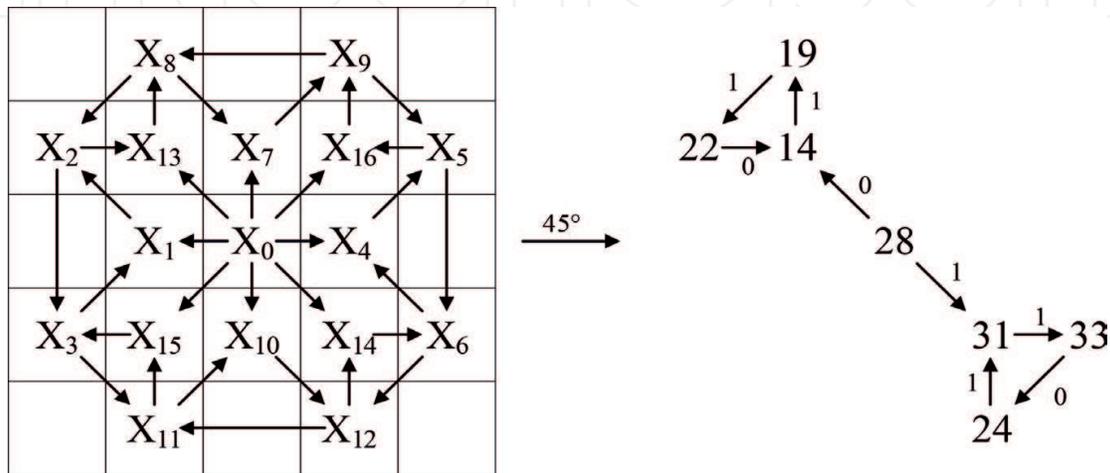


Figure 3. Graph structure of MOW-SLGS in the direction of 45°.

3. Proposed method

The characteristic value of the target pixel extracted by LGS and MOW-SLGS algorithm are in the neighborhood of 3×4 and 5×5 . It can be seen that these two algorithms do not consider all the surrounding pixels in the neighborhood. This leads to the lack of information about the surrounding pixels, which will have an impact on the recognition rate.

In order to solve this problem, we propose a LCGS algorithm by using the full information of the surrounding pixels. In such way, LCGS can get more representative features for the target pixels, thus, improve the recognition rate.

The way to extract the characteristics of the face image with LCGS algorithm is illustrated in Figure 4.

In Figure 4, the center pixel X_0 is taken as the target pixel in the neighborhood of 3×3 . The remaining pixels are represented by X_1 to X_8 respectively. We put all the pixels in accordance with $X_0 \rightarrow X_8 \rightarrow X_1 \rightarrow X_0, X_0 \rightarrow X_2 \rightarrow X_3 \rightarrow X_0, X_0 \rightarrow X_4 \rightarrow X_5 \rightarrow X_0, X_0 \rightarrow X_6 \rightarrow X_7 \rightarrow X_0$ to compare the value of the pixel. If the value of the pixel is larger in the direction of the arrow, we set it to 1; otherwise, we set it to 0. Finally, we obtain 12 binary numbers in sequence from $X_0 \rightarrow X_8$ to $X_7 \rightarrow X_0$. The decimal format of these 12 binary values is the characteristic value. A specific example is shown in Figure 5.

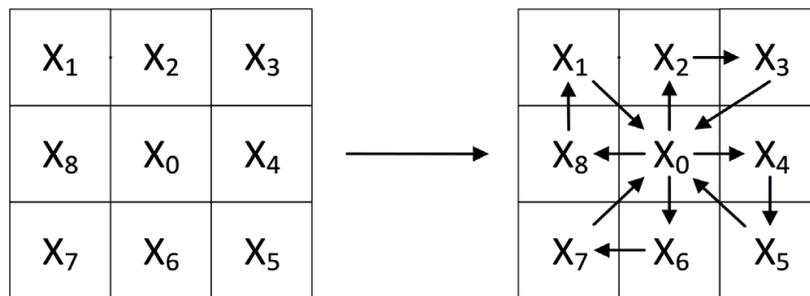


Figure 4. LCGS operator.

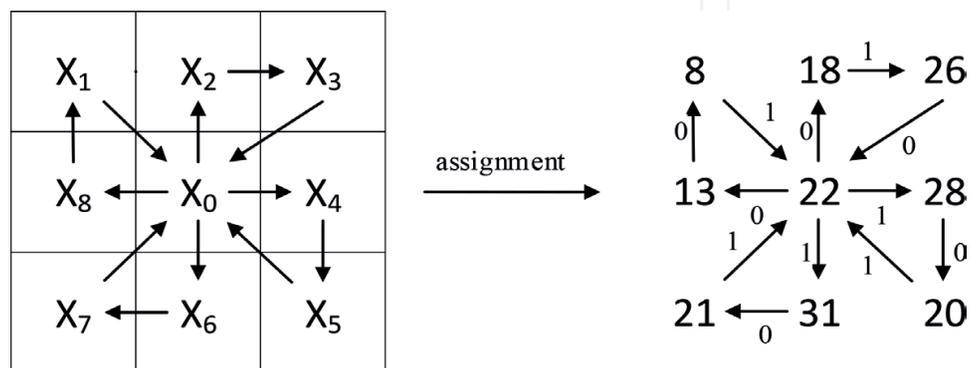


Figure 5. Template of LCGS operator.

In **Figure 5**, we calculate the characteristic value according to our proposed LCGS algorithm. The target pixel value is 22. The value of target pixel is 22. We compare the value of pixels in according to the order of $X_0 \rightarrow X_8 \rightarrow X_1 \rightarrow X_0 \rightarrow X_2 \rightarrow X_3 \rightarrow X_0 \rightarrow X_4 \rightarrow X_5 \rightarrow X_0 \rightarrow X_6 \rightarrow X_7 \rightarrow X_0$ and get the binary values as 001010101101. It is converted to the decimal number, that is, $F(X_0) = 0 \times 2048 + 0 \times 1024 + 1 \times 512 + 0 \times 256 + 1 \times 128 + 0 \times 64 + 1 \times 32 + 0 \times 16 + 1 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1 = 685$, which gives the final feature values for the target pixel.

In the LCGS algorithm, we define the neighborhood of 3×3 along the traditional way. It is effective to calculate the characteristic values of the target pixel by considering all the surrounding pixels in the neighborhood. Thus, it makes the final recognition rate improved.

4. Experiments and analysis

In order to investigate the reliability of LCGS algorithm, we use the AR physical occlusion database and the ORL simulation occlusion database to conduct the experiments. The AR face database consists of 126 people, a total of 3276 images, 26 pictures per person. The 26 pictures are collected in two different time periods, including changes in light, occlusion, and so on. The shelter is divided into scarves and sunglasses. The ORL Face Database was created by the University of Cambridge AT & T Lab, which contains 40 people, 10 images per person, and a total of 400 face images. We use a baboon image to randomly block the original image for the occlusion simulation. Some examples of AR database and ORL database are shown in **Figures 6** and **7**.



Figure 6. Examples of AR face database.

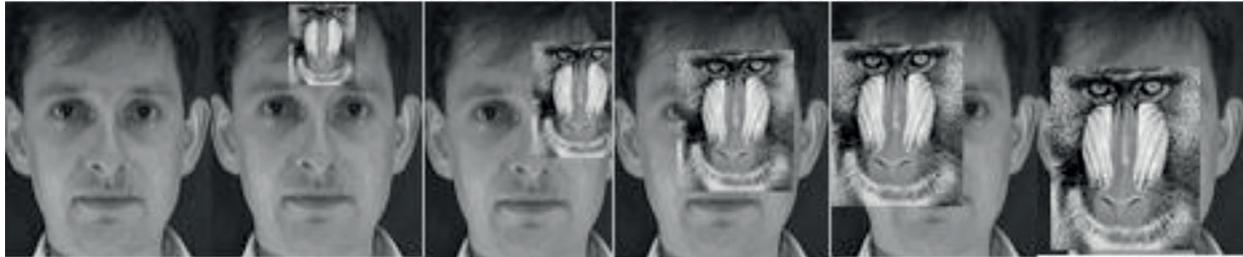


Figure 7. Examples of ORL face database.

4.1. Dimensionality reduction using principal component analysis (PCA)

The LCGS algorithm is used to extract the features of a face image, and the dimension of the feature matrix is usually very large, which makes it difficult to use the classifier to train and test. Therefore, we adopt a state-of-the-art method Principal Component Analysis (PCA) [11, 12] to reduce the dimension after feature extraction. For the implementation of PCA, we set the principal component contribution rate as 0.95.

4.2. Training and classification using extreme learning machine (ELM)

In the stage of image classification, we employ the Extreme Learning Machine (ELM) [13, 14] to train and classify the data. This algorithm is a supervised learning algorithm for single-hidden layer feed-forward network (SLFN). The main idea of ELM is to determine the number of hidden layer neurons, then perform random assignment on the input weights and hidden layer bias, and finally the output layer weights can be directly calculated by the least squares method. The entire learning process is completed at once, without iteration, so its learning speed is very fast. Based on the extensive experimental experience, we set the number of hidden layer nodes to 2000.

4.3. The experimental results

4.3.1. Experimental results on AR face database

In this section, we compared the LCGS algorithm with the traditional LGS, SLGS and MOW-SLGS algorithms in terms of recognition rate on AR face database. Hundred people in the database are selected as our experimental data, where half of men and women. The training set selects unobstructed face images, seven images per person, and a total of 700 images. The test set is divided into three parts. The shelter in test data set 1 is a scarf, three images per

Test set	LGS	SLGS	MOW-SLGS	LCGS
Test set 1 (scarf)	0.8367	0.8847	0.8967	0.9127
Test set 2 (sunglasses)	0.8970	0.9413	0.9380	0.9533
Test set 3 (scarf + sunglasses)	0.8620	0.9147	0.9203	0.9295

Table 1. Recognition rates of different algorithms on the AR database.

person, a total of 300 images. The shelter in test data set 2 is sunglasses, three images per person, a total of 300 images. The shelter in test set 3 is a mixture of scarves and sunglasses, each with six images, a total of 300 images. The results of the recognition rates are shown in **Table 1**.

Through **Table 1**, we can see that recognition rate achieved by LCGS algorithm is higher than other algorithms for all these three test data set. In particular, when the shelter of the test set is a scarf, the recognition rates of the LGS, SLGS, and MOW-SLGS algorithms are 83.67, 88.47, and 89.67%. Moreover, the recognition rate of the LCGS algorithm is 91.27%, which well demonstrates the advantages of our proposed method in the occluded images with a scarf. We also find that the recognition rate for test set 1 is the lowest among the three test data sets. This is due to the occlusion caused by the scarf almost accounted for 20% of the image. It has a great effect on the feature extraction of the whole image.

4.3.2. Experimental results on ORL face database

We performed an analog occlusion experiment on the ORL face database, and we chose LGS, SLGS, and MOW-SLGS to compare our proposed algorithm, LCGS. We used a baboon picture to block the original face picture randomly. We set the occlusion area as 10%, 20%, 30%, 40%, and 50% of the original image. We set up three training sets. The first group selected six images per person, a total of 240 images. The second group selects seven images per person, a total of 280 images. The third group selects eight images per person, a total of 320 images. Corresponding, the test set is also divided into three groups. The test set 1 is four images per person, a total of 160 images. The test set 2 is three images per person, a total of 120 images. The test set 3 is two images per person, a total of 80 images. The results are shown in **Table 2**, **Figure 8**, and **Table 3**.

Occlusion area (%)	LGS	SLGS	MOW-SLGS	LCGS
10	0.5425	0.5731	0.6375	0.6713
20	0.5156	0.5531	0.6188	0.6275
30	0.4587	0.5300	0.5587	0.6019
40	0.3625	0.4450	0.4731	0.5063
50	0.2863	0.3606	0.3775	0.4044

Table 2. Recognition rates of different algorithms on test set 1 of the ORL database.

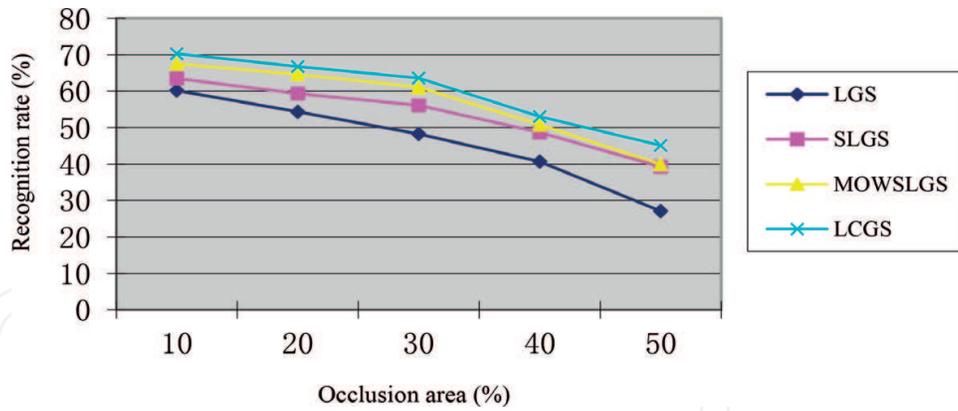


Figure 8. Recognition rates of different algorithms on test set 2 of the ORL database.

Occlusion area (%)	LGS	SLGS	MOW-SLGS	LCGS
10	0.5700	0.5988	0.6975	0.7013
20	0.5362	0.5838	0.6663	0.6738
30	0.4900	0.5688	0.5925	0.6200
40	0.4025	0.4763	0.5275	0.5463
50	0.2888	0.4113	0.3825	0.4188

Table 3. Recognition rates of different algorithms on test set 3 of the ORL database.

From **Table 2**, **Figure 8**, and **Table 3**, we can clearly see that the recognition rate of the LCGS algorithm is higher than that of the conventional algorithms. In the test set 1, the recognition rates of the LGS, SLGS, and MOW-SLGS algorithms are 45.87%, 53.00% and 55.87%, respectively, when the occlusion area is 30%, and the recognition rate of the LCGS algorithm is 60.19%.

4.3.3. 10-fold cross-validation

In order to further verify the accuracy of the algorithm, we also conducted 10-fold cross-validation. We chose the ORL database and randomly masked only one piece of data for testing. The comparison algorithms are LGS, SLGS, and MOW-SLGS. The result is shown in **Table 4**.

Through the experimental results, as shown in **Table 4**, we can see that LCGS algorithm performs better than other algorithms. The recognition rates of the LGS, SLGS, and MOW-SLGS algorithms are 49.00%, 53.00%, and 55.25%, respectively, when the occlusion area is 40% while the recognition rate of the LCGS algorithm is 61.75%.

4.3.4. Comparison of the processing time

During the experiments, we compared the processing time for the same image required by the LGS, SLGS, MOW-SLGS, and our proposed LCGS algorithm. The result is shown in **Table 5**.

Occlusion area (%)	LGS	SLGS	MOW-SLGS	LCGS
10	0.6900	0.7150	0.7875	0.7825
20	0.6850	0.6900	0.7475	0.7575
30	0.5650	0.6400	0.6875	0.6925
40	0.4900	0.5300	0.5525	0.6175
50	0.3625	0.4675	0.4275	0.5200

Table 4. 10-fold cross-validation experiment results for different algorithms.

Algorithms	LGS	SLGS	MOW-SLGS	LCGS
Processing time (second)	0.2886	0.2895	1.0093	0.4493

Table 5. The processing time of different algorithms.

Through **Table 5**, we can see that the processing time required for one image by the LCGS algorithm is 0.4493 seconds. Although the required time is higher than that of LGS and SLGS, it is significantly less than the time required by the MOW-SLGS. The reason is that the MOW-SLGS algorithm calculates the feature values of the four directions around the target pixel. Each direction is located on both sides of the target pixel, that is, a total of eight sets of feature values are calculated. The proposed LCGS algorithm only calculates the four sets of feature values based on the full usage of the surrounding pixels. Therefore, MOW-SLGS's processing time is about twice as high as LCGS.

5. Conclusion

In this chapter, we proposed a LCGS algorithm and applied it to face recognition with occlusion. LCGS makes full use of the texture features of the surrounding pixels in the 3×3 neighborhood. It makes up for the shortcomings of LGS and MOW-SLGS algorithm where the information around the target pixel is not sufficient. The characteristics of the target pixel value using LCGS algorithm is easier to recognize. Therefore, it improves the recognition rates with occlusion. Through the experiments on the AR and ORL database, we demonstrated that the LCGS algorithm is superior to the traditional algorithm in the recognition rate of face images with occlusion, and the time consumption is lower than the MOW-SLGS algorithm.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No.61502338 and No. 61502339, the 2015 key projects of Tianjin science and technology

support program No. 15ZCZDGX00200, the Project of Scientific Research Plan of Tianjin Education Committee NO.2017KJ034, and Tianjin Food Safety & Low Carbon Manufacturing Collaborative Innovation Center.

Author details

Jucheng Yang, Lingchao Zhang, Meng Li, Tingting Zhao*, Yarui Chen, Jianzheng Liu and Na Liu

*Address all correspondence to: tingting@tust.edu.cn

College of Computer Science and Information Engineering, Tianjin University of Science and Technology, Tianjin, China

References

- [1] Kumar A, Zhou Y. Human identification using finger images. *IEEE Transactions on Image Processing*. 2012;**21**(4):2228-2244
- [2] Ahonen T, Hadid A, Pietikainen M. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2006;**28**(12):2037-2041
- [3] Yu H, Yang J. A direct LDA algorithm for high-dimensional data—With application to face recognition. *Pattern Recognition*. 2001;**34**(10):2067-2070
- [4] Ojala T, Pietikainen M, Harwood D. Performance evaluation of texture measures with classification based on Kullback discrimination of distributions. In: *Pattern Recognition, 1994. Proceedings of the 12th IAPR International Conference on Conference A: Computer Vision & Image Processing*. Vol. 1. Jerusalem, Israel: IEEE; 1994. pp. 582-585
- [5] Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2002;**24**(7):971-987
- [6] Tan X, Triggs B. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*. 2010;**19**(6):1635-1650
- [7] Li X-X, Liang R. A review for face recognition with occlusion: From subspace regression to deep learning. *Chinese Journal of Computers*. 2017;**40**:82
- [8] Abusham EE, Bashir HK. Face recognition using local graph structure (LGS). In: *International Conference on Human-Computer Interaction*. Berlin, Heidelberg: Springer; 2011. pp. 169-175
- [9] Abdullah MFA, Sayeed MS, Muthu KS, Bashier HK, Azman A, Ibrahim SZ. Face recognition with symmetric local graph structure (slgs). *Expert Systems with Applications*. 2014;**41**(14):6131-6137

- [10] Dong S, Yang J, Chen Y, Wang C, Zhang X, Park DS. Finger vein recognition based on multi-orientation weighted symmetric local graph structure. *KSII Transactions on Internet and Information Systems*. 2015;**9**(10):4126-4142
- [11] Luo Y, Zhang T, Zhang Y. A novel fusion method of PCA and LDP for facial expression feature extraction. *Optik-International Journal for Light and Electron Optics*. 2016;**127**(2):718-721
- [12] Turk M, Pentland A. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*. 1991;**3**(1):71-86
- [13] Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: Theory and applications. *Neurocomputing*. 2006;**70**(1):489-501
- [14] Zong W, Huang G-B. Face recognition based on extreme learning machine. *Neurocomputing*. 2011;**74**(16):2541-2551

IntechOpen

