



Local Binary Pattern Algorithm with Weight Threshold for Image Classification

Zexi Xu, Guangyuan Qiu, Wanying Li, Xiaofu He^(✉), and Shuang Geng

College of Management, Shenzhen University, Shenzhen, China

Abstract. Image classification has attracted the attention in many research field. As an efficient and fast image feature extraction operator, LBP is widely used in the Image classification. The traditional local binary pattern (LBP) algorithm only considers the relationship between the center pixel and the edge pixel in the pixel region, which often leads to the problem of partial important information bias. To solve this problem, this paper proposes an improved LBP with threshold, which can significantly optimize the processing of texture features, and also be used to address the problems of multi-type image classification. The experimental results show that the algorithm can effectively improve the accuracy of image classification.

Keywords: Image classification · Feature extraction · Dimensionality reduction · BP network

1 Introduction

In the modern era of data explosion, people are exposed to a great variety of pictures. The first step to process the data of these pictures is to classify them [1, 2]. With the continuous development of computer technology, artificial intelligence and other fields, the number and types of data are increasingly diverse. Therefore, image classification has attracted people's attention. Image classification usually includes image information acquisition, image preprocessing, image detection, feature extraction and dimensionality reduction and then the image classification is eventually achieved.

LBP (local binary pattern) is a practical and popular texture feature extraction operator. it is widely used for its principle is easy to understand and the extracted texture features are obvious,. Ahonen et al. [3] proposed a face image representation method based on local binary pattern (LBP) texture features. Guoying Zhao et al. [4] improved the extended VLBP(volume local binary patterns) algorithm to deal with dynamic features and applied it in expression recognition. Wei Li et al. [5] proposed a framework of LBP based image local feature extraction and two-level fusion with global Gabor feature and original spectral feature to classify hyperspectral images with high spatial resolution, and achieved good results.

However, the traditional LBP Operator only considers the relationship between the gray value of the center pixel and the neighboring pixel when obtaining the eigenvalues, which is easily affected by the change of gray value and reduces the recognition rate. An

improved LBP algorithm introduces the concept of threshold. Firstly, the gray values of the neighborhood pixel and the center pixel are obtained by calculation, and the influence on the threshold is further calculated. On this basis, the LBP algorithm of weighted threshold UM proposed in this paper introduces the weight U to the ordinary threshold M, so as to reduce the influence of the maximum and minimum value in the calculation of threshold M, and make the threshold M more distinguishable. The algorithm also optimizes the field of face recognition. When extracting the features of mouth, eyebrow, eye and other parts, we can increase the weight U appropriately to make the features of these parts more obvious.

2 Literature

2.1 LBP Algorithm

LBP (Local Binary Pattern) is first put forward by T. Ojala et al. in 1994 for texture feature extraction, it has the significant advantages in terms of rotation invariance and gray invariance, as well as the extraction of the local texture feature of the image.

Description of LBP Features

The original LBP [6, 7] is based on the Central Pixel of the window as the threshold, comparing the gray values of 8 adjacent pixels in the 3 * 3 window. When the surrounding pixels are larger than the Central Pixel, the position of the central point is marked as 1, otherwise it is marked as 0, eight adjacent pixels can have a string of eight binary digits. Then, the LBP value of the Pixel in the center of the window is calculated and used to reflect the texture information of the region [8].

The image encoding is shown in Eq. (1):

$$P_{ALL} = \sum_{l=1}^N (P_l) \quad (1)$$

$$s(g_p - g_c) = \begin{cases} 1, & g_p \geq g_c \\ 0, & g_p < g_c \end{cases} \quad (2)$$

Where: g_c stands for the gray value of the center point; g_p stands for the gray value of each point in the 8 neighborhood; $s()$ is a symbolic function;

Figure 1 shows a 3 × 3 local binary mode encoding scheme. With 83 as the center pixel, binary processing is carried out in a clockwise direction from the top left. If the gray value is greater than 83, it is marked as 1, otherwise, it is marked as 0, after that the LBP values for the center pixel are achieved.

2.2 KPCA

The Principle Component Analysis (PCA) is widely used in various scenarios which is a mathematical transformation method that interprets most of the information of the original variable by combining a few linear combinations of the original variables and transforming multiple dimensions into a few main components that are not related

88	77	58
109	84	125
99	79	92

1	0	0
1		1
1	0	1

binary system: 10011011 decimal system: 155

Fig. 1. Schematic diagram of the original LBP

to each other. However, the principle component analysis method has its limitations in processing nonlinear data. As an alternative, KPCA method is used to solve the nonlinear problem [9].

Kernel Principal Component Analysis (KPCA) is a nonlinear PCA method that first projects data dimensions into the feature space through nonlinear transformations and analyzes the main components in the feature space. As shown in the following illustration: In a two-tier spherical data, PCA classification is not effective, while KPCA is able to classify data (Fig. 2).

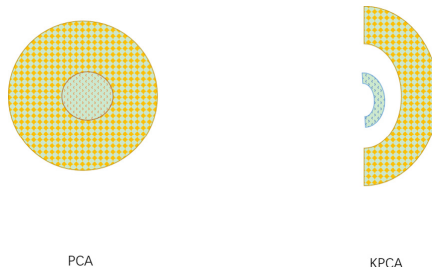


Fig. 2. The difference between PCA and KPCA

2.3 BP Neural Network

BP neural network is a multi layer neural network, it is one of the most widely used neural network, it uses the gradient descent to make the actual output and the expected output error minimum [10].

The BP network adds several layers of neurons between the input and output layers, which become implicit layers without direct connection to the outside world, but their state changes can influence the relationship between input and output. Each layer has several nodes. BP neural network has a strong nonlinear mapping capability and flexible network structure. The performances of different networks with different structures are also distinct [11].

The training of BP neural network is to update its two parameters. The training process of BP neural network includes: forward-looking calculation, input data processing and obtaining actual value;

Reverse calculation: the error signal between the actual output and the expected output is transmitted backwards along the network connection path. The weights and bias of each neuron are corrected to minimize the error signal.

3 Proposed Method

3.1 Improved LBP Algorithm (1): LBP with Threshold Value M

The original LBP solely compares the gray value between the center pixel and the neighborhood, without considering the relationship between them, so it often omits some important feature information.

An improved LBP is proposed to solve this problem. Firstly, a threshold value M [12] is introduced to calculate the difference between the gray values of the neighborhood pixels and the center pixel by adding the absolute values and then taking the average value. The average value is taken as the threshold value. If it is greater than the threshold value, it is 1; if it is less than or equal to the threshold value, it is 0. The LBP value calculated in this way takes into account not only the function of the center pixel, but also the relationship between the neighborhood.

First, the gray value of the center pixel is subtracted from the gray value of the adjacent pixel, and then added to get the threshold value M. the formula (3) is as follows:

$$M = \frac{1}{p} \sum_{i=0}^{p-1} |g_i - g_c| \tag{3}$$

Second, the gray value of GP and GC of the center pixel are subtracted from the absolute value in a certain order to compare with the threshold value M, greater than is 1, less than is 0, as shown in formula (4)

$$s(g_p - g_c) = \begin{cases} 1, & |g_p - g_c| > M \\ 0, & |g_c - g_c| \leq M \end{cases} \tag{4}$$

Then the binary code is converted into decimal number, and the formula (5) is as follows:

$$LBP^*_{P,R} = \sum_{I=1}^{P-1} S(g_p - g_c) 2^I \tag{5}$$

Where: p stands for the number neighborhood pixels, where p is 8; gc stands for the gray value of the center point; gp stands for the gray value of each point in the 8 neighborhood; s () is a symbolic function;

An improved LBP coding process is shown in Fig. 3.

88	77	58
109	84	125
99	79	92

0	0	1
1		1
0	0	0

M: 16 binary system: 00110001 decimal system: 49

Fig. 3. Schematic diagram of the LBP with Threshold Value M

3.2 Improved LBP Algorithm (2): LBP with Weight Threshold UM

The simple introduction of a threshold M ignores the importance of the key areas. In the image classification, there are often several core areas of images. Among these core areas, the introduction of a weight threshold called UM can effectively increase the degree of differentiation. The weight refers to the proportion of the neighborhood excluding the maximum value when calculating the threshold value M. The larger the weight is, the less the threshold is affected by the maximum value, which makes it more stable and distinguishable.

Firstly, the absolute value is obtained by subtracting the gray value of the center pixel from the gray value of the adjacent pixel, as is shown in formula (6) (7).

$$\text{Max} = \text{Max} (g_i - g_c), I = 0, 1, \dots, p - 1 \tag{6}$$

$$\text{Min} = \text{Min} (g_i - g_c), I = 0, 1, \dots, p - 1 \tag{7}$$

Then we change the ratio of the minimum to the maximum of the original M to optimize the threshold M, as shown in Eq. (8).

$$M = \frac{U * \sum_i^{p-1} |g_i - g_c| - (U - 1) * (\text{Max} + \text{Min})}{6U + 2} \tag{8}$$

$$s(g_p - g_c) = \begin{cases} 1, & |g_p - g_c| > M \\ 0, & |g_c - g_c| \leq M \end{cases} \tag{9}$$

Where P is the number of adjacent pixels, gc is the gray value of the central point, gi is the gray value of the current central point, and U is the quality;

The operation is shown in the figure below (Fig. 4):

The following figure shows the LBP feature map after LBP extraction of an image (Fig. 5):

88	77	58
109	84	125
99	79	92

0	0	1
1		0
0	1	1

$U=2, UM=15$ binary system: 00110011 decimal system: 51

Fig. 4. Schematic diagram of the LBP with Weight Threshold UM

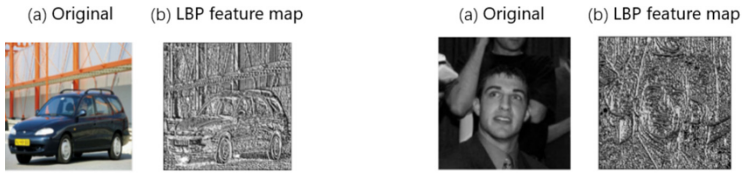


Fig. 5. Feature image of LBP

4 Experiment and Results

4.1 Experiment

This paper is based on the dataset that comes from Imagenet, the largest database of image recognition in the world: colorful pictures of different categories are respectively placed under Car, Dog, Face and Snake. There are about five thousand images in each category. Among them, the dataset is divided into the training set and the test set in a ratio of 7:3 (Fig. 6).



Fig. 6. Pictures of four training sets

The main processes includes:

- ① Data preprocessing: In this paper, all the color images of the training set are preprocessed and transformed into 256×256 Gy images.
- ② Feature extraction: In this paper, LBP basic mode, LBP threshold M and LBP weight threshold UM are used for feature extraction successively, and their statistical histogram is used as feature vector [13].

- ③ Data dimensionality reduction: Since the picture belongs to nonlinear data, KPCA is used to process these data, so that it can be greatly reduced in dimensionality.
- ④ BP neural network training: The feature vectors obtained from dimension-reduction are stored in trainInputs, and the image categories corresponding to the training set are stored in trainTargets. A BP neural network with four neurons is constructed for training.
- ⑤ Result evaluation: ACC, Micro-F1, Macro-F1 [14] and other accuracy coefficients are used for evaluation (Fig. 7).

$$P_I = \frac{TP_I}{TP_I + FP_I}, (I = 1, \dots, N) \tag{10}$$

$$R_I = \frac{TP_I}{TP_I + FN_I}, (I = 1, \dots, N) \tag{11}$$

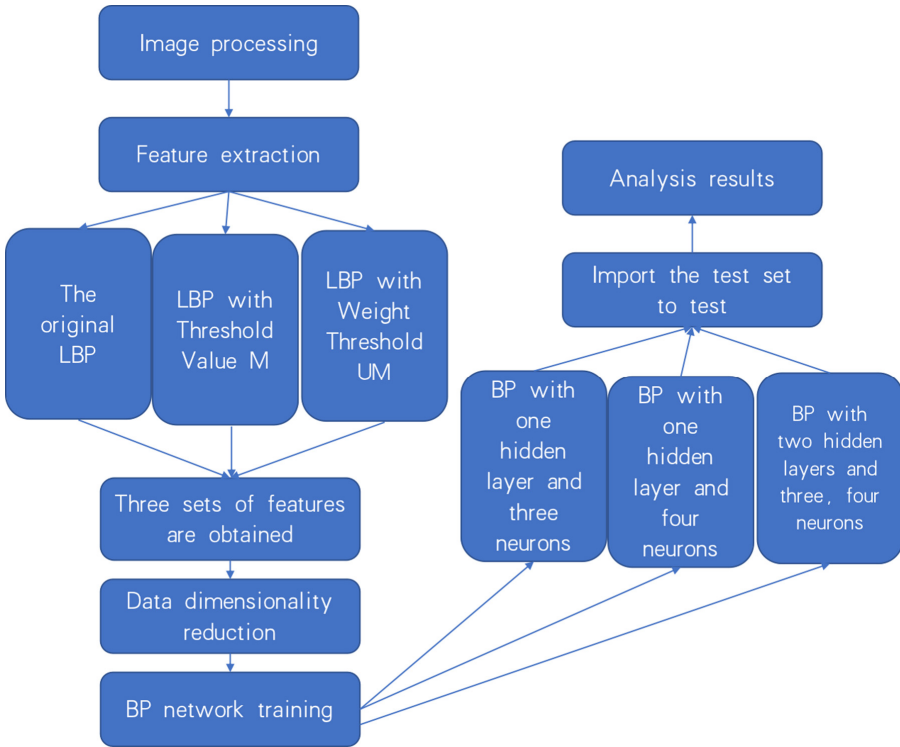


Fig. 7. Experimental process

Where: TP represents true positive, which means the prediction is positive, and it's actually positive; FP represents false positive, which means the prediction is positive, actually negative; FN represents false negative, which means the prediction is negative, actually positive; N represents the number of categories of pictures.

F1 score is an index, which is used to measure the accuracy of binary classification model in statistics. It takes into consideration the accuracy and recall of the classification

model. The F1 score can be perceived as a weighted average of the accuracy rate and recall rate of the model, with the maximum value of 1 and the minimum value of 0.

Generally speaking, Micro-F1 and Macro-F1 belong to two types of F1 indicators. These two F1 are used as evaluation indexes in multi-class classification tasks, and they are two different ways to calculate the mean value of F1. There are differences between the calculation methods of Micro-F1 and Macro-F1, and the results are also slightly different.

Calculation method of Micro-F1: first calculate the total precision and recall of all categories, and then calculate the Micro-F1:

$$P_{ALL} = \sum_{I=1}^N (P_I) \quad (12)$$

$$R_{ALL} = \sum_{I=1}^N (R_I) \quad (13)$$

$$Micro_F1 = \frac{2 * P_{ALL} * R_{ALL}}{P_{ALL} + R_{ALL}} \quad (14)$$

Calculation method of Marco-f1: average precision and recall of all categories, and then calculate the Macro-F1:

$$FL_I = \frac{2 * P_I * R_I}{P_I + R_I} \quad (15)$$

$$Macro_F1 = \frac{\sum_{I=1}^N (FL_I)}{N} \quad (16)$$

4.2 Comparative Analysis of Experimental Results

300 images of each of these four kinds of images are selected as the test set, and the test is conducted according to three kinds of feature extraction methods. The test results are as follows (Tables 1, 2 and 3 and Figs. 8, 9 and 10):

Table 1. The result of BP network test of three neurons

Kind of LBP	ACC	Macro-F1	Micro-F1
1. The original LBP	0.8401	0.7995	0.8401
2. LBP with Threshold Value M	0.8517	0.8322	0.8517
3. LBP with Weight Threshold UM, U = 1	0.8750	0.8419	0.8750
4. LBP with Weight Threshold UM, U = 2	0.8625	0.8632	0.8625
5. LBP with Weight Threshold UM, U = 3	0.8642	0.8655	0.8642
6. LBP with Weight Threshold UM, U = 4	0.8500	0.8506	0.8500

Table 2. The result of BP network test of four neurons

Kind of LBP	ACC	Macro-F1	Micro-F1
1. The original LBP	0.8514	0.8519	0.8514
2. LBP with Threshold Value M	0.8762	0.8544	0.8762
3. LBP with Weight Threshold UM, U = 1	0.8912	0.8777	0.8912
4. LBP with Weight Threshold UM, U = 2	0.8821	0.8835	0.8821
5. LBP with Weight Threshold UM, U = 3	0.8864	0.8873	0.8864
6. LBP with Weight Threshold UM, U = 4	0.8650	0.8675	0.8650

Table 3. The result of BP network test of three, four neurons

Kind of LBP	ACC	Macro-F1	Micro-F1
1. The original LBP	0.8382	0.6892	0.8382
2. LBP with Threshold Value M	0.8673	0.8413	0.8673
3. LBP with Weight Threshold UM, U = 1	0.8874	0.8345	0.8874
4. LBP with Weight Threshold UM, U = 2	0.8545	0.8588	0.8545
5. LBP with Weight Threshold UM, U = 3	0.8745	0.8733	0.8745
6. LBP with Weight Threshold UM, U = 4	0.8403	0.8420	0.8403

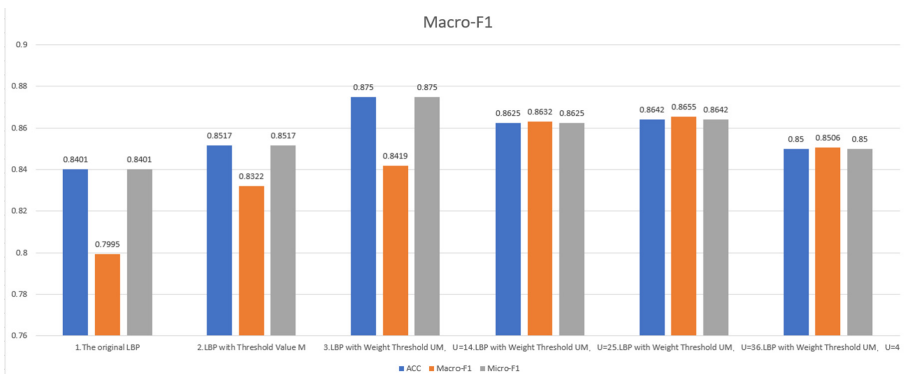


Fig. 8. The result of BP network test of three neurons

As can be seen from the above three histograms, when we use LBP with weight threshold UM, in the same BP network structure, ACC of the original LBP is 0.8401; ACC of LBP with threshold value m is 0.8517, and LBP with weight threshold UM is 0.8517. UM's ACC reaches 0.8750, which increases by approximately 3%. When the total number of pictures climbs, this seemingly small difference can significantly

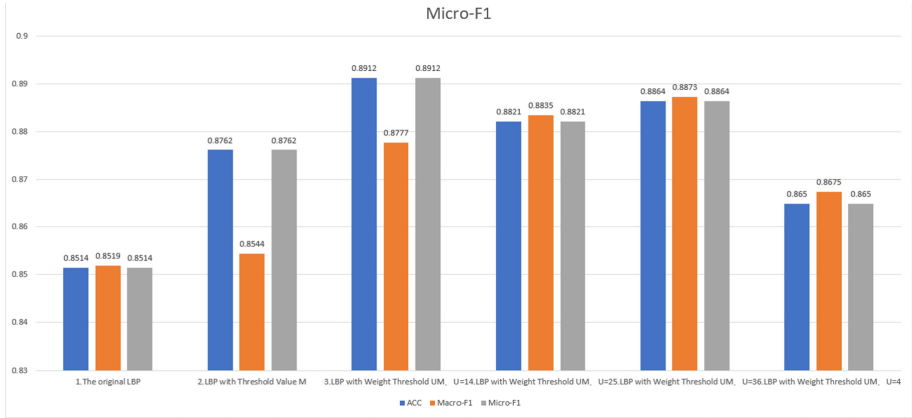


Fig. 9. The result of BP network test of four neurons

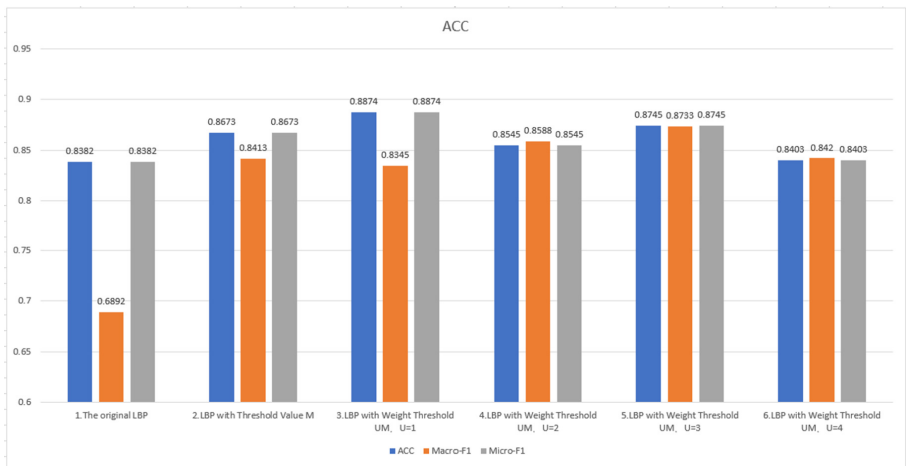


Fig. 10. The result of BP network test of three, four neurons

improves the degree of accuracy of the test. The change of Macro-F1 can also reflect the superiority of the improved UM. Next, we discuss the BP structure and LBP algorithm.

In the structure of BP neural network, in the above test, the increase of the number of hidden layers is not equal to the increase of the accuracy of the experiment, but if the numbers of hidden layers are the same, increasing the number of neurons is conducive to the optimization of the test results and the improvement of the accuracy of the experiment. In three different BP neural networks, it is proved that the LBP model with threshold UM is the best.

In the basic mode of LBP, LBP with threshold M and LBP with weight threshold UM, it can be concluded that the LBP algorithm with weight threshold UM is the best based on the comparison of these three LBP feature extraction methods. The main reason

is that the appropriate weight U greatly optimizes the threshold M , which improves the degree of discrimination for this kind of image, and thus improves the accuracy.

Finally, according to the table above, while $u = 2$ is sometimes slightly better, $U = 1$ or 3 is significantly better when combined. The idea of using weight to emphasize the importance of a particular region as an important basis for judging an image is innovative.

5 Conclusion

The original LBP Algorithm only calculates the difference of gray value between Central Pixel and adjacent Pixel, and only considers the result analysis of gray value between them, and does not consider the direct relationship between them. This may result in the loss of local features of some important information, which affects the final classification result. This problem can be solved by introducing threshold into LBP, but too high threshold will lead to insufficient differentiation. Therefore, it is suggested to raise the threshold of Body weight (UM). By setting appropriate weight U , the threshold M is more reasonable and the effect is more obvious. According to the experimental results, this method effectively improves the classification efficiency.

The idea of using weight to emphasize the importance of a particular region as an important basis for judging an image is innovative. However, the optimal value of U is different in different research areas, which requires more experiments to obtain the optimal weight of U . In addition, when we apply this weight idea to face recognition and other fields, its advantages will be more confirmed. Due to the limited space of the experiment, I did not apply it to the special field of face recognition, but I want to put forward an idea: when we carry out face recognition, we can increase the weight U of the nose, mouth, eyes and other areas with the most personal characteristics in the face image, so that when we recognize the image, we pay more attention to these high weight U blocks, so that the recognition effect can be more ideal.

Acknowledgement. This study is supported by National Natural Science Foundation of China (71901150, 71702111, 71971143), the Natural Science Foundation of Guangdong Province (2020A151501749), Shenzhen University Teaching Reform Project (Grants No. JG2020119).

References

1. Bin, J., Jia, K., Yang, G.: Research progress of facial expression recognition. *Comput. Sci.* **38**(4), 25–31 (2011)
2. Yan, O., Nong, S.: Facial expression based on combined features of facial action units recognition. *Chin. Stereol. Image Anal.* **16**(1), 38–43 (2011)
3. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(12), 2037–2041 (2006). <https://doi.org/10.1109/TPAMI.2006.244>
4. Zhao, G., Pietikainen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**(6), 915–928 (2007). <https://doi.org/10.1109/TPAMI.2007.1110>

5. Li, W., Chen, C., Hongjun, S., Qian, D.: Local binary patterns and extreme learning machine for hyperspectral imagery classification. *IEEE Trans. Geosci. Remote Sens.* **53**(7), 3681–3693 (2015). <https://doi.org/10.1109/TGRS.2014.2381602>
6. Zhou, Y., Wu, Q., Wang, N.: Discriminant complete local binary pattern facial expression recognition. *Comput. Eng. Appl.* **53**(04), 162–169 (2017)
7. Werghi, N., Tortorici, C., Berretti, S., Del Bimbo, A.: Boosting 3D LBP-based face recognition by fusing shape and texture descriptors on the mesh. *IEEE Trans. Inf. Forensics Secur.* **11**(5), 964–979 (2016). <https://doi.org/10.1109/TIFS.2016.2515505>
8. Li, C., Liu, Y., Li, C.: Application of PCA and KPCA in comprehensive evaluation. *Yibin College J.* **10**(12), 27–30 (2010)
9. Wu, J. (ed.): *The Theory and Practice of Integrated Automation System of Water Resources Engineering*. China Water and Hydropower Press (2006)
10. Lu, M., Zhou, H.: An improved multi-scale local binary pattern expression recognition method. *Hebei Agric. Mach.* **10**, 65–69 (2017)
11. Lei, J., Lu, X., Sun, Y.: Expression recognition based on improved local binary pattern algorithm (10), 36–37 (2018)
12. Xin, W., Sheng, Z., Zhu, Y.: *Intelligent Troubleshooting Technology: MATLAB Application*. Beijing University of Aeronautics and Astronautics Press, Beijing (2015)
13. Li, L.: Expression recognition algorithm based on LBP hierarchical feature **32**, 733–735 (2013)
14. Zhang, H.: An overview and research perspective of local binary pattern. *J. Image Signal Process.* **05**(3), 121–146 (2016)