Multimedia Tools and Applications (2022) 81:5389–5405
https://doi.org/10.1007/s11042-021-11776-1

Extended complete local binary pattern for texture classification

Zeng Qiang[2,3] · Adu Jianhua[1] · Sun Xiaoya[1] · Hong Sunyan[1]

Received: 27 June 2021 / Revised: 21 September 2021 / Accepted: 26 November 2021 /
Published online: 21 December 2021

Abstract

In this paper, an extended complete LBP (ELBP) for texture classification is proposed, in which the local feature vectors are composed of the ratio of the central pixel and its neighborhood pixels to a specific threshold. ECLBP_C represents the gray level of the image, which is obtained by comparing the center pixel with the global threshold. ECLBP_S and ECLBP_M represent the symbol component and the magnitude component of the 3-neighbor region of the center pixel respectively, which are obtained by calculating two binary codes using the original LBP algorithm for the 3-neighbor region of the center pixel. In order to make the proposed algorithm scalable, in addition to the 3-neighbor pixels of the central pixel, the proposed algorithm use the center pixel as the center, r as the radius in the circle with α as the filter's radius to generate extended binary coding, such as ECLBP_ES_r,α ECLBP_EM_ r,α. In order to describe the local region feature vector in detail, specified ECLBP_ES_r,α and ECLBP_EM_r,α can be obtained by defining the number of extensions according to actual needs, and then established and concatenated all ECLBP gray histograms for statistics. In the experimental part, we analyze the performance of the proposed algorithm in detail, and prove that the algorithm has good scalability and robustness. The experimental results show that the classification accuracy of the proposed algorithm is up to 99% after 3 expansions in Table 2. The source codes of the proposed algorithm can be downloaded from https://github.com/zenqiang/ECLBP.git.

Keywords Texture classification · ECLBP · Local binary pattern

✉ Adu Jianhua
adujh@126.com

[1] Kunming University, Kunming 650214, China

[2] Software Department, Chengdu University of Information Technology, Chengdu 610225, China

[3] College of Cybersecurity, Sichuan University, Chengdu 610065, China

1 Introduction

Texture is an important feature of the image, and texture classification is one of the important research topics of computer vision. Texture classification plays an important role in target recognition, medical image analysis and understanding, biometrics, content-based image retrieval, remote sensing and industrial detection. Early texture feature extraction algorithms were numerous, but most of them were based on a common assumption that texture images were acquired under ideal conditions. In the mid-1990s, Zhang et al. began to advocate the study of invariant texture classification methods [39]. Texture classification presents a new situation, texton theory [14] was proposed by Julesz et al., Bag of Textons (BoT) texture classification pattern had been onto stage, which is an important word package pattern in the field of computer vision. In 2004, Csurka introduced the concept of the word package into the field of computer vision [3], and began a large amount of research work focused on the word package pattern. Leung et al. studied the identification of texture images acquired under different imaging angles and illumination conditions, and they proposed the LM filter algorithm [16]. Varma et al. improved the LM filter algorithm and proposed a local texture feature extraction algorithm MR8 with rotation invariance [37]. In addition, Varma et al. proposed a simple patch feature to represent texture features in order to prove that the filtering algorithm dominates texture analysis [38].

Ojala et al. proposed a method for classifying texture images with rotation invariance using a local binary pattern (LBP) histogram [28]. On account of the LBP algorithm is simple, easy to implement, certain brightness invariance, it is favored by a wide range of researchers. As one of the most successful feature description operators, LBP algorithm has successfully demonstrated its efficiency in face recognition [1], dynamic texture recognition [40] and shape recognition [12].

The LBP algorithm is simple and effective, but it has its limitations, researchers have done a lot of research on the LBP algorithm, and various LBP variant algorithms have been presented. In order to improve the accuracy of LBP algorithm for texture recognition, many extended and improved algorithms have been proposed, such as Complete Local Binary Pattern (CLBP) proposed by Guo et al. [5], Extended Local Binary Pattern (ELBP) proposed by Liu et al. [24], Dominant Local Binary Pattern (DLBP) proposed by Liao et al. [19], Discriminative Complete Local Binary Pattern (disCLBP) proposed by Y. et al. [7], Pairwise Rotation Invariant Cooccurrence Local Binary Pattern (PRICoLBP) proposed by Qi et al. [31]. And These algorithms increase the dimension of LBP algorithm to improve the texture classification accuracy, but all of them are sensitive to the noise. At the same time, due to the increase in feature dimensions, the computational complexity is increased. In order to improve the robustness of the LBP algorithm to noise, some improved algorithms have been proposed, such as Local Ternary Pattern (LTP) proposed by Tan et al. [34], Median Binary Pattern (MBP) [8] proposed by Hafiane et al., Local Phase Quantization (LPQ) proposed by Ojansivu etc. [29], Robust Local Binary Pattern (RLBP) proposed by Chen et al. [2], Noise Resistant Local Binary Pattern (NRLBP) proposed by Ren et al. [32], Noise Tolerant Local Binary Pattern (NTLBP) proposed by Fathi et al. [4], Fuzzy Local Binary Pattern (FLBP) proposed by Iakovidis et al. [13], Median Robust Extended Local Binary Pattern (MRELBP) proposed by Liu et al. [18].Q Kou propose a new local texture descriptor, the principal curvatures based LBP (PCLBP) [15], Z Pan proposed a new method named scale-adaptive local binary pattern (SALBP) [30], and R Tekin proposed four novel, simple and robust approaches, which are left to right local binary patterns (LBPL L2R), top to down local binary patterns (LBP T2D), cube surface local binary pattern (LBP Surfaces), and cube diagonal local binary pattern (LBP Diagonal), in order to exact texture features in color images [35]. MH Shakoor fused the LDT features to the features of rotation invariant LBP and local variance (VAR) to provide a rich set of robust-to-noise features [33]. M Hazgui proposed a genetic programming (GP)-based method that combines of the two well-known features: histograms of oriented gradients and local binary patterns [10]. A new local structure descriptor based on just noticeable difference (JND) for texture classification is proposed by exploring the spatial and relative intensity correlations among local neighborhood pixels [25]. Compared with the LBP algorithm, these algorithms are more robust to image noise, but their anti-noise ability is still not satisfactory. In addition, many scholars analyze and improve the LBP algorithm. Guo et al. believe that the LBP algorithm has the disadvantage of losing global spatial information, so they improved the LBP algorithm and put forward the LBPV algorithm, which improved the accuracy of texture classification by 10 percentage points [6]. Hegenbart et al. analyzed the sensitivity of LBP algorithm to affine transformation and proposed a local binary mode based on scale and rotation invariance [11]. Zhou et al. used the principle of LBP algorithm to introduce Huffman coding rules and proposed a Haffman LBP algorithm for face recognition [41]. Moscoso et al. proposed an edgeLBP algorithm to capture geometric patterns on the surface grid [26]. Further, Xin Liao et al. introduced a deep learning framework for image detection in literature [20], which not only obtains significant detection performance but also can distinguish the order in some cases that previous works were unable to identify. Xin Liao et al. also introduced the features of texture image and color image respectively in literature [21] and [22].

In this paper, we improved the CLBP algorithm and proposed an Extended Complete Local Binary Pattern (ECLBP). The proposed ECLBP algorithm further improves and extends on the basis of the CLBP algorithm, inheriting the advantages of CLBP and making texture classification more accurate. In literature [5], Guo et al. found that the texture information contained in the sign component is much larger than the magnitude component and the introduction of the magnitude component allows the feature detection operator to distinguish the change of the brightness level. The proposed algorithm improves and extends CLBP and has all the advantages of the CLBP. Furthermore, ECLBP algorithm adds a filter and replace the single pixel with the filtered grayscale result of the window region, which improves the robustness of the proposed algorithm to image noise. Like CLBP, the local feature vector in the ECLBP algorithm is composed of the ratio of the center pixel and its neighboring pixels to a specific threshold. The ratio of the center pixel to the global threshold represents the gray level of the image, denote as ECLBP_C. In this proposed algorithm, the uniform rotation invariant LBP is adopted as the binary encoding method in the 3 neighboring pixels of the central pixel, and the binary encoding can be divided into sign binary encoding and magnitude binary encoding, denote as ECLBP_S and ECLBP_M respectively. In addition, with reference to the extended characteristics of the Freak [36] feature describe operator, the proposed algorithm adds Extended Sign binary encoding and Extended Magnitude binary encoding. The Extended binary takes the center pixel as the center, r as the radius and α as the filter radius respectively, to generate specified extended sign binary codes and extended magnitude binary codes, denote as ECLBP_ES_r,α and ECLBP_EM_r,α respectively. Finally, combining all ECLBP_C, ECLB_S, ECLBP_M, ECLBP_ES, ECLBP_EM gray-level histograms to generate the final ECLBP feature vector for texture feature analysis or texture classification.

In this paper, we propose a robust, extensible, flexible and effective texture feature extraction algorithm. Experiments demonstrate that the performance of this algorithm is better than the previous variant LBP algorithm, and the accuracy of texture classification reaches more than 99%. Below is a summary of our main contributions.

- The proposed algorithm improves and extends CLBP and it has expansibility, flexibility, brightness invariance and robustness.
- Furthermore, the proposed algorithm adds a filter and replace the single pixel with the filtered grayscale result of the window region, which improves the robustness of the proposed algorithm to image noise.
- In the experiment, the proposed algorithm is analyzed in detail and is compared with LBP and its representative variants: LTP [34], CLBP [5] and MRELBP [18]. On some integrated texture dataset, the experimental results show that the classification accuracy of the proposed algorithm is up to 99%.
- Different from MRELBP [18] algorithm, the proposed algorithm takes time performance consumption into account in the design, the proposed algorithm is more concise in the feature coding, and greatly reduces the calculation time consumption while ensuring high precision.

This paper is organized as follows. The following section briefly explains the principle of LBP [28] and MRELBP [18], and Section 3 discusses in detail the implementation principle of the proposed algorithm. In Section 4, the results and analysis of experiments are presented. Finally, our conclusions are given in Section 5.

2 Related work

The LBP algorithm can mainly measure and extract texture information of local neighborhoods in grayscale images. In the original LBP operator, a 3 * 3 fixed rectangular region, a total of 9 pixels, are selected and the gray value of the 8 pixels in the edge of the rectangular region are compared with the intermediate pixel, which is greater than or equal to the central pixel, the value is represented by 1, otherwise it is represented by 0. Finally, according to the clockwise serial encoding, an encoding value of the LBP operator can be obtained. In order to make the LBP can encode any size region, Ojala etc. improved the original LBP and proposed the circular LBP(P,R) [28]:

$$\begin{aligned} LBP_{P,R} &= \sum_{p=0}^{p-1} s\left(g_p - g_c\right)2^p \\ \mathrm{s(x)} &= \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \end{aligned} \tag{1}$$

where g_p represents the gray value of the neighborhood pixel, g_c represents the gray value of the center pixel, P represents the count of the encoding, R represents the neighborhood radius. At the same time, pixels that are not in the image grid can be obtained using interpolation. Suppose that the coordinates of g_c is (0,0), the coordinates of g_p is $(\mathrm{R}\cos(2\pi p/P), \mathrm{R}\sin(2\pi p/P))$.

In order to increase the robustness of rotation invariant, Ojala etc. proposed rotation invariant LBP:

$$LBP_{P,R}^{riu} = \min\left\{ROR\left(LBP_{P,R}, i\right) | i = 0, 1, \dots, P - 1\right\} \tag{2}$$

where ROR(x, i) indicates that x is sequentially shifted to right by i bits, and then the minimum value is taken as the unique feature encoding [28].

For LBP(P,R), there are a total of 2^p sequence of 0 and 1 with a high dimension. In order to reduce the dimension, Ojala proposed uniform local binary pattern and pointed out that under natural visible light, 90% of the image texture pattern belong to the uniform pattern [28]. In the actual image, most LBP patterns only contain no more than two transitions from 1 to 0 or from 0 to 1. Therefore, Ojala defines the uniform pattern as follows: when the cyclic binary number corresponding to the LBP only contain no more than two transitions from 0 to 1 or from 1 to 0, the binary encoding corresponding to the LBP is called an equivalent mode.

$$\begin{aligned} &\mathrm{U}\left(LBP_{P,R}\right) = \left|\mathrm{s}\left(g_{p-1} - g_c\right) - \mathrm{s}(g_0 - g_c)\right| + \sum_{p=0}^{P-1} s\left(g_p - g_c\right) - s\left(g_{p-1} - g_c\right) \\ &LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s\left(g_p - g_c\right)\mathrm{U}\left(LBP_{P,R}\right) \leq 2 \\ P + 1 else \end{cases} \end{aligned} \tag{3}$$

where U represents the number of encoding transitions from 0 to 1 or from 1 to 0. For example, 00000000 (0 of the transitions), 00000111 (1 from 0 to 1 of the transitions), 10001111 (from 1 to 0, and then from 0 to 1, there are 2 of the transitions) are uniform pattern. Except uniform pattern, the other patterns are called the mixed pattern. For example, 10010111 (4 of the transitions). The riu2 represents rotation invariant uniform pattern adopted by LBP and there are P + 1 modes. The 9 encoding modes are combined with rotation-invariant LBP and uniform mode LBP are shown in Fig. 1.

For LBP algorithm is simple, easy to implement and has certain brightness invariance, it has been widely favored by researchers. MRELBP designs a new feature classification descriptor for the traditional LBP algorithm, which is sensitive to noise and unable to obtain macroscopic information. Different from other LBP variant algorithms, MRELBP algorithm improves ELBP [23] algorithm and proposes to use the median of regional image instead of single pixel intensity of the original image, which makes the algorithm has stronger anti-noise ability, and has good brightness invariance, rotation invariance and robustness. MRELBP algorithm represents: Images are normalized to zero mean and unit variance; the standard (riu2) encoding scheme can be used; and the joint histogramming of RELBP_CI, RELBP_NI$^{riu2}_{r,p}$ and RELBP_NI$^{riu2}_{r,p}$ is used to represent a texture image. This new descriptor is referred to as RELBP$^{riu2}_{r,p}$ [18].

In the formula, each component of MRELBP is expressed as RELBP_CI RELBP_NI RELBP_RD:

1) Center pixel representation:

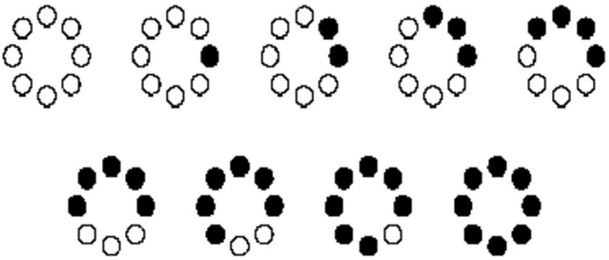


Fig. 1 $LBP^{riu2}_{8,r}$: the hollow circle represents 0, and the solid circle represents 1

$$\mathrm{RELBP}_{\mathrm{CI}(\mathrm{x}_c)} = \mathrm{s}(\phi\left(X_{c,w}\right) - \mu_w) \tag{4}$$

where x_c represents the center pixel, the filter $\phi()$ to $X_{c,w}$ is applied to the calculation results, w means the size of w × w sliding window, and μ_w represents the mean of $\phi\left(X_{c,w}\right)$ over the whole image.

2) Neighbor representation:

$$\begin{gathered}\mathrm{RELBP}_{\mathrm{NI}r,p}\left(x_c\right) = \frac{1}{p}\sum_{n=0}^{p-1}\mathrm{s}\left(\phi\left(X_{r,p,w_r,n}\right) - \mu_{r,p,,w_r}\right)2^n \\ \mu_{r,p,,w_r} = \frac{1}{p}\sum_{n=0}^{p-1}\phi\left(X_{r,p,w_r,n}\right)\end{gathered} \tag{5}$$

where $X_{r,p,w_r,n}$ denotes a patch of size $w_r \times w_r$ centered on $x_{r,p,n}$.

3) Radial difference representation:

$$\mathrm{RELBP}_{RDr,r-1,p,w_r,w_{r-1}}\left(x_c\right) = \frac{1}{p}\sum\nolimits_{n=0}^{p-1}\mathrm{s}\left(\phi\left(X_{r,p,w_r,n}\right) - \phi\left(X_{r-1,p,w_{r-1},n}\right)\right)2^n \tag{6}$$

where $X_{r,p,w_r,n}$ and $X_{r-1,p,w_{r-1},n}$ denote the patches centered the neighboring pixel $x_{r,p,n}$ and $x_{r-1,p,n}$ respectively.

MRELBP algorithm uses the response value calculated by the filter window to replace the single pixel, which has stronger anti-noise ability. Meanwhile, the introduction of RELBP_NI and RELBP_RD enhances the feature extraction performance of the algorithm, making MRELBP algorithm achieve great success in the application of texture classification. However, MRELBP also has its own defects, mainly as follows.

1. MRELBP calculates RELBP_CI, RELBP_NI and RELBP_RD components for each extension point. The algorithm is acceptable when the number of extension layers is not high, but when the number of extension layers exceeds a certain number, the time consumption of the algorithm increases geometratically.
2. MRELBP calculates the final feature vector for each component according to the same weight. When the number of extended layers is large, more noise will be introduced, which is not conducive to the final classification.
3. As the design of MRELBP algorithm is more complex, it is not flexible enough for simple texture image classification.

In order to overcome these shortcomings, in this paper we refer to CLBP algorithm and design the center gray level, sign component and magnitude component to generate the final texture feature vector. In the proposed algorithm, it is different from MRELBP, in each layer extension the proposed algorithm only need to compute the expansion of the specific component coding, which makes the algorithm have bigger promotion in the expansion flexibility. At the same time, more macro information can be introduced into feature vectors to extract complex texture features on the basis of performance overhead. Since the extended component of the proposed algorithm is generated only for its own features, the introduction of noise is avoided, and the final classification accuracy will not be affected even if the extended component of higher level numbers is introduced. Secondly, the time complexity of the proposed algorithm is O(n), and the algorithm speed is higher than that of MRLBP algorithm. In the third part of this paper, we give a detailed derivation of the proposed algorithm.

3 ECLBP algorithm

The features in ECLBP are divided into three categories: ECLBP_C representing the gray level of the image; ECLBP_S-M representing the sign component and the magnitude component; ECLBP_ES-M representing the extended sign component and the extended magnitude component.

The LBP algorithm and many variant LBP algorithms are sensitive to image noise since all of them use only a single pixel as the sample point for encoding. Referring to the Freak feature description operator [36] and the BRISK feature description operator [17], they all use a Gaussian filter response value of a specific window for feature description, which is much better than using only a single pixel as a sampling point.

The proposed ECLBP algorithm improves the CLBP algorithm and refers to the pattern of the Freak algorithm [36]. ECLPB algorithm inherits the brightness invariance and rotation invariance of CLBP algorithm, and has strong anti-noise ability due to the introduction of filtering window. Secondly, because the design of each component of ECLPB algorithm is simpler than that of MRELBP, it has a great breakthrough in time performance, and the extended component of ECLBP algorithm is generated based on the characteristics of the extension point itself, avoiding the introduction of noise, and the improved pattern is shown in Fig. 2. In Fig. 2, the extended regions of the ECLBP overlap each other, which makes the correlation of the grayscale response values of the neighborhood edges more closely [36]. Filters for overlapping regions can choose Gaussian filter or Median filter. Gaussian filter is one of most common linear smoothing filter, and it is widely applied to eliminate noise in image processing. The median filter is a non-linear digital filter. Because it is non-linear, the median filter is especially useful for anti-noise effects of speckle noise, salt and pepper noise. In Section 4, the experiments of texture classification using Gaussian filtering and median filtering are compared, and the final results are analyzed in detail.

Figure 3 shows the generation process of the sign component (ECLBP_S) and the magnitude component (ECLBP_M) in the ECLBP algorithm. Each edge pixel in the

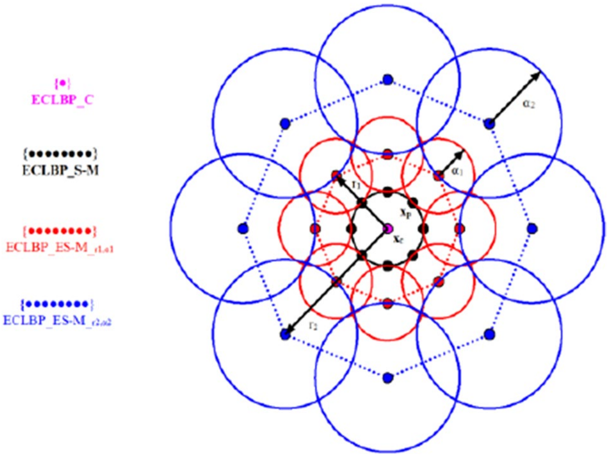
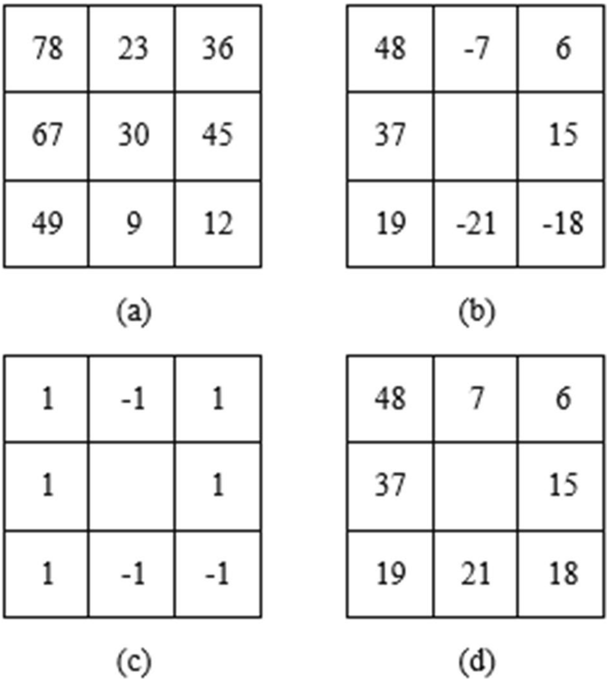


Fig. 2 The pattern of the ECLBP algorithm: where x_c represents the center pixel gray value response, x_p represents the 3-neighbor pixels gray value response, r represents the distance of the extended feature and the center pixel, α represents the window size of filter in the extended region; it is able to change the value of r and α according to the actual demand for a further expansion



78	23	36
67	30	45
49	9	12

(a)

48	-7	6
37		15
19	-21	-18

(b)

1	-1	1
1		1
1	-1	-1

(c)

48	7	6
37		15
19	21	18

(d)

Fig. 3 The generate process of the sign component and magnitude component: (**a**) represents 3-neighbor pixels gray value; (**b**) represents the difference between the neighborhood pixels and the center pixel; (**b**) can be represents as (**c**) × (**d**); (**c**) represents the sign component, Change the value encoded as -1 to 0; (**d**) represents the magnitude component

3-neighbor is respectively performed a difference operation with the central pixel, and the obtained result is combined clockwise to obtain the sequence A[6,15,-18,-21,19,37,48,-7]. Split the sign of the sequence with the value to obtain the sequence B[1,1,-1-,1,1,1,-1] and the sequence C[6,15,18,21,19,37,48,7]. The sequence B and the sequence C are processed again to obtain the binary encoding of the sign component and the magnitude component in the ECLBP. The sign component replaces 1 in sequence C with 0. By processing sequence B, the magnitude component compares the average value of sequence B with each value and codes the one greater than the average value as 1, otherwise it codes as 0.

Given the central pixel x_c, ϕ represents the filter window response, the extended radius is r, and the extended region filter size is α, the calculation process for ECLBP_C, ECLBP_S, ECLBP_M, ECLBP_ES_r, and ECLBP_EM_r in ECLBP is as follows.

A. Gray level of the center pixel:

$$ECLBP\_C_{P,R} = s(x_c - \mu)$$
$$\mathrm{s(x)} = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \tag{7}$$

where P represents the count of encoding, R is the distance of the 3-neighborhood pixel from the center pixel, μ represents the grayscale average of the image, and s denotes the encoding rules. In this paper, P = 8, R = 1 in the experiment.

B. Sign component

$$ECLBP\_S_{P,R} = \sum\nolimits_{p=0}^{p-1} s\left(x_p - x_c\right)2^p \tag{8}$$

where x_p represents the edge sample pixel of the 3-neighborhood pixel around the center pixel. The ECLBP_S is essentially the original LBP binary code.

C. Magnitude component

$$\begin{aligned} &ECLBP_{MP,R} = \textstyle\sum_{p=0}^{P-1} s\left(m_p - \lambda\right)2^p \\ &m_p = |x_p - x_c| \\ &\lambda = \frac{\sum_p^{P-1}|x_p - x_c|}{P} \end{aligned} \tag{9}$$

where m_p represents the absolute difference between the neighborhood pixel and the center pixel, λ is the average value of the grayscale response magnitude of the neighborhood pixel around the center pixel.

D. Extended sign component:

$$\begin{aligned} &\mathrm{ECLBP\_ES\_r}, \alpha = \textstyle\sum_{p=0}^{P-1} s\left(\phi\left(x_{r,p}, \alpha\right) - \delta\right)2^p \\ &\delta = \frac{\sum_{p=0}^{P-1}\phi\left(x_{r,p}, \alpha\right)}{P} \end{aligned} \tag{10}$$

where $\left(x_{r,p}, \alpha\right)$ represents the gray response value of $x_{r,p}$ with the filter α × α, $x_{r,p}$ represents the sample pixel of in extended area, δ represents the average value of the filtering window response at the sampling point.

E. Extended magnitude component:

$$\begin{aligned} &\mathrm{ECLBP\_EM\_r}, \alpha = \textstyle\sum_{p=0}^{P-1} s\left(em_p - \delta\right)2^p \\ &em_p = |\phi\left(x_{r,p}, \alpha\right) - \delta| \end{aligned} \tag{11}$$

where em_p represents the absolute difference between the extended pixel of the filter response and the gray mean value of the extended pixel of the filter response.

After obtaining ECLBP_C, ECLBP_S, ECLBP_M, ECLBP_ES, and ECLBP_EM respectively, and then all components are serially connected to generate a gray histogram for classification. In addition, the number of ECLBP_ES and ECLBP_EM can be expanded according to actual needs. In the experiments in this paper, the classification accuracy can be more than 99% after 4 times of expansion. The flow chart of the proposed algorithm is shown in Fig. 4.

4 Experiment

To verify the effectiveness of the algorithm, two integrated texture datasets, Outex [27] and KTHTIPS [9], were used for experiments. The Outex dataset contains 24 types of texture images, each contains 3 different lighting conditions, 9 rotation angles, and no effects of viewpoint changes and scale changes. The KTHTIPS dataset contains 10 types of material texture images, each contains 3 different lighting conditions, 3 different poses and 9 different scales. The detail of the experimental dataset is shown in Table 1. Figure 5 shows some texture images in the KTHTIPS database and Fig. 6 shows some texture images in the Outex_TC12 database. The source code of the proposed algorithm can be downloaded from https://github.com/zenqiang/ECLBP.git.

The ECLBP algorithm is an improvement and extension of the CLBP algorithm and belongs to a variant LBP algorithm. Therefore, we use Outex_TC10 and Outex_TC12 in

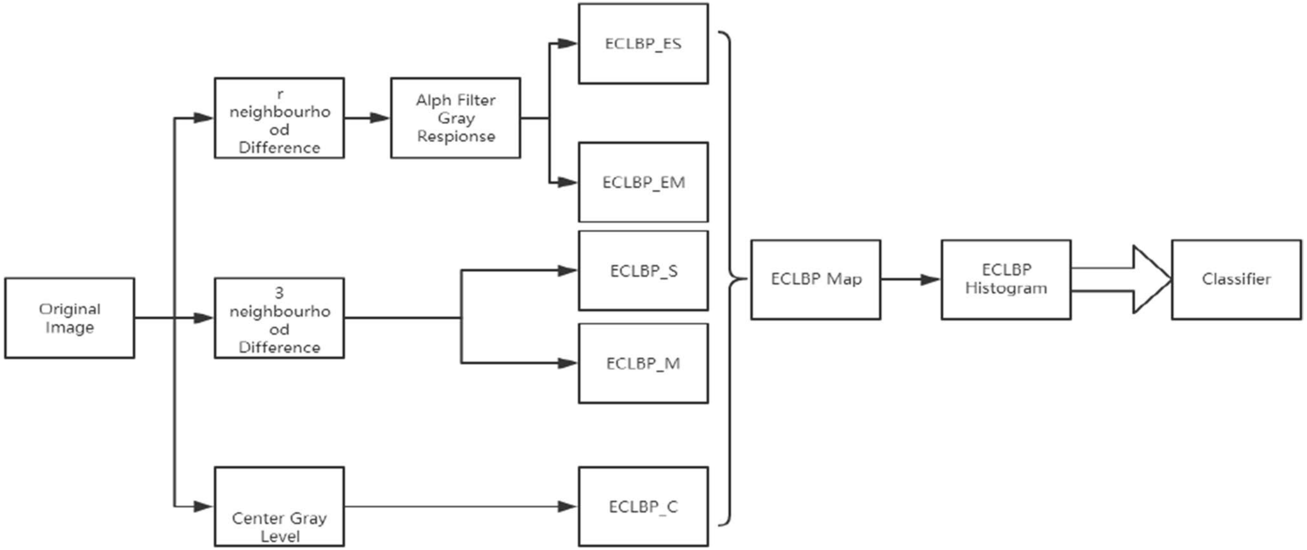


Fig4 The flow chart of ECLBP algorithm

Table 1 Summary Texture Dataset Used In Our Experiments. θ_1(00,05,10,15,30,45,60,75,90)

Texture Dataset	Texture Classes	Rotations(°)	Resolutions(dpi)	Number of train images	Number of test images
Outex_TC10	24	θ_1	100	480	3840
Outex_TC11	24	00	100,120	480	480
Outex_TC12	24	00	100	480	480 or 960
KTHTIPS	10	00	200	360	450

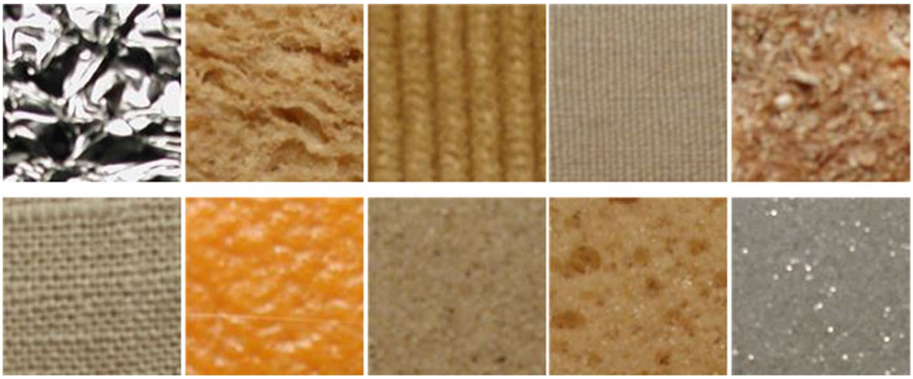


Fig. 5 Texture images in KTHTIPS dataset

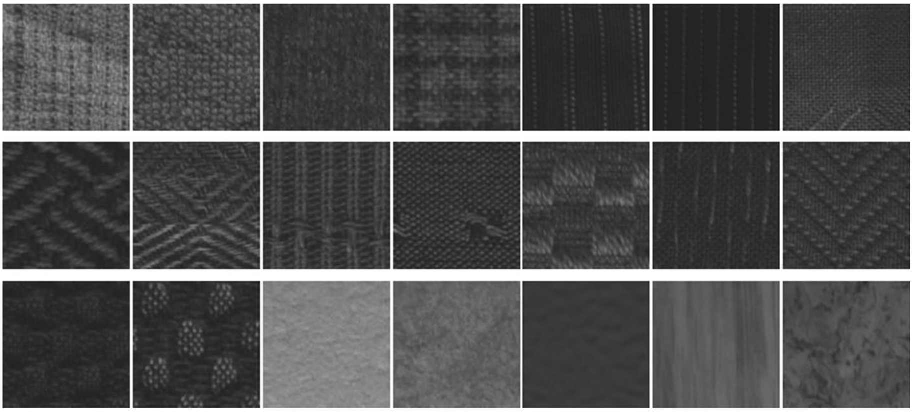


Fig. 6 Texture images in Outex_TC12 dataset

the Outex dataset to experiment. We compared the LBP algorithm [28], LTP algorithm [34], CLBP algorithm [5], MRELBP algorithm [18] by using Outex_TC10 and Outex_TC12 data set under the same experimental conditions. In this paper, the classification accuracy is calculated by the following formula.

$$\text{accuracy} = \frac{correct\_classification}{numberofallimages} \tag{12}$$

The LBP algorithm [28] is the most primitive local binary mode and is the theoretical basis of all variant LBP algorithms. In the LBP algorithm, a neighborhood sample pixel of a specific radius is compared with a central pixel to obtain a binary code, and the final binary code is calculated according to the rotation invariant and equalization mode.

The LTP algorithm [34] divides the local difference into a ternary pattern using a threshold to improve the LBP algorithm, and divides the ternary mode into two LBPs: positive LBP and negative LBP.

The CLBP algorithm [5] is a complete local binary mode. On the basis of the original LBP algorithm, the gray level of the central pixel and the neighborhood pixel magnitude component are added, which makes the CLBP algorithm have better performance in texture classification.

MRELBP algorithm [18] is a novel algorithm for texture image classification. The MRELBP algorithm extends MRELBP_CI, MRELBP_NI, and MRELBP_RD referring to the Freak algorithm, and the MRELBP algorithm uses median filter to improve the robustness to image noise.

In order to verify the scalability of the ECLBP algorithm, in the same experimental environment, we designed the experiment to use ECLBP_C | ECLBP_S-M | ECLBP_ES-M_4,3, ECLBP_C | ECLBP_S-M | ECLBP_ES-M_6,5, ECLBP_C | ECLBP_S-M | ECLBP_ES-M_8,7 ECLBP_S | ECLBP_S-M | ECLBP_ES-M_10,9, ECLBP_C | ECLBP_S-M | ECLBP_ES-M_12,11 separately classify the texture image, and then classify the texture image again by concatenating the various extension components.

Figure 7 shows the ECLBP feature gray-level histogram, in which the gray histogram extracted from the image using the extended ECLBP algorithm. When extracting features using the ECLBP algorithm, each sub-area of size 16×16 is encoded once for each image, and finally these encodings are combined in series to form a final ECLBP feature histogram.

Table 2 shows the classification result of the current main LBP-based texture classification algorithms and the proposed algorithm in Outex_TC10 and Outex_TC12 dataset. From Table 2, it can be found that after three extensions, the classification accuracy of the proposed algorithm has reached more than 99% for the Outex_TC10 data set, and after five extensions, the classification accuracy reaches the highest 99.67%. For the Outex_TC12 data set, tl84 and horizontal represent texture images acquired under different light sources. In the Outex_TC12 dataset under the tl84 light source, the proposed algorithm is accurate to 99.20% after 5 extensions, and the Outex_TC12 dataset under the horizontal source the proposes algorithm achieves the highest classification accuracy 99.56% after 4 expansions. However after the fifth expansion, the classification accuracy decreased by 0.11%.

In Section 3, we introduced that the ECLBP algorithm add a filter and replace the single pixel with the filtered grayscale result of the window region. There are two alternative filters: Gaussian filter and Median filter. Gaussian filter can be used as a representative of linear filter, and median filter can be used as a representative of nonlinear filter. For the Outex_TC11 data set, we designed an experiment to verify the extended components in the ECLBP algorithm using Gaussian filter and Median filter, respectively.

Table 3 shows the experimental results of the proposed algorithm for verifying the extended components using Gaussian filter and Median filter, respectively. From Table 3, it can be found that for the ECLBP algorithm, after each extension the classification result of the median filter is several percentage points higher than the classification result of the Gaussian filter. For the proposed algorithm, after using the Gaussian filter and the Median filter and completing 2 extensions, the classification accuracy both are over 97%. After 3 extensions, the ECLBP algorithm using the median filter achieves the highest classification

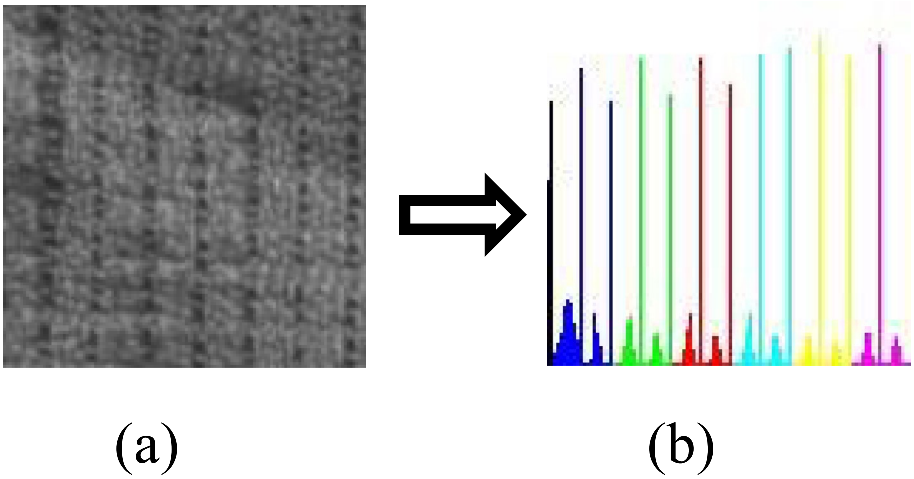


Fig. 7 ECLBP gray histogram of 5 extension: (**a**) represents the original texture image; (**b**) represents the histogram of (**a**), where the gray-level histogram of different color represents the different component; the black represents ECLBP_C, only 2 positions in the gray-level histogram; the blue represents ECLBP_S-M component, 18 positions in the gray-level histogram; the green represents ECLBP_ES-EM_4,3 component, what represents the first extension, 18 positions in the gray-level histogram; the red represents ECLBP_ES-EM_6,5 component, what represents the second extension,18 positions in the gray-level histogram; the sapphire represents ECLBP_ES-EM_8,7 component, what represents the third extension, 18 positions in the gray-level histogram; the yellow represents ECLBP_ES-EM_10,9 component, what represents the fourth extension, 18 positions in the gray-level histogram; the purple represents ECLBP_ES-EM_12,11 component, what represents the fifth extension, 18 positions in the gray-level histogram

Table 2 Classification rate (%) on Outex_TC10 and Outext_TC12 using different extended S-M

Method	Outex_TC10	Outex_TC12		Average	Time (second/ image)
		tl84	Horizon		
LBP	91.36	81.25	82.56	85.06	0.0124
LTP	96.11	85.20	85.87	89.06	0.0625
CLBP	96.56	90.30	92.29	93.05	0.0354
MRELBP	99.58	98.82	99.35	99.25	0.2645
ECLBP_4,3	98.24	95.41	96.32	96.66	0.0469
ECLBP_6,5	97.36	95.00	95.41	95.92	0.0389
ECLBP_8,7	98.59	96.56	96.87	97.34	0.0406
ECLBP_10,9	97.32	96.77	97.56	97.22	0.0399
ECLBP_12,11	96.89	95.93	95.94	96.25	0.0458
ECLBP_4,3|6,5	98.97	95.52	98.21	97.57	0.0697
ECLBP_4,3|6,5|8,7	99.21	97.08	98.56	98.28	0.0899
ECLBP_4,3|6,5|8,7|10,9	99.45	98.23	99.56	99.08	0.0972
ECLBP_4,3|6,5|8,7|10,9|12,11	99.67	99.20	99.45	99.44	0.1156

Table 3 Classification rate (%) on Outex_TC11 using different filter. Gaussian noise, $\sigma = 5$, none blur

Method	Gauss	Median
ECLBP_4,3	90.42	91.25
ECLBP_6,5	85.42	90.13
ECLBP_8,7	88.33	90.83
ECLBP_10,9	88.75	89.17
ECLBP_12,11	87.50	89.17
ECLBP_4,3|6,5	97.08	97.50
ECLBP_4,3|6,5|8,7	97.40	98.65
ECLBP_4,3|6,5|8,7|10,9	98.13	98.54
ECLBP_4,3|6,5|8,7|10,9|12,11	98.44	98.65

accuracy of 98.65%, while the classification accuracy of the ECLBP algorithm using Gaussian filter is 97.40%. When the ECLBP algorithm performing 4 extensions, the classification accuracy of the ECLBP algorithm using the Median filter is reduced by 0.11 percentage points, but still higher than the 98.13% of the ECLBP algorithm using Gaussian filter. After 5 expansions, the classification accuracy of the ECLBP algorithm using the median filter again reached 98.65%.

We also designed an experiment to verify the classification effect of the proposed algorithm with different numbers of training samples in the KTHTIPS data set. With the KTHTIPS dataset, we separately verified the classification results when the number of training samples for each texture image type was 10, 20, 30 and 40, and the experimental results are shown in Table 4. As seen from Table 4, when the number of training samples for each texture image type is 10, the highest accuracy of classification is only 76.36%. When the training sample is increased to 20, it can be found that after the proposed algorithm is only extended once, the highest classification accuracy has reached 82.22%. Once again, the training sample is increased to 30, and after 1 extension the highest classification accuracy of the proposed algorithm is up to 88.89%, which is 5.47 percentage points higher than the highest classification accuracy of the training sample at 20. When the training sample was increased to 40, the highest classification accuracy of the proposed algorithm was up to 95.06% and the lowest was 91.36%. When the extension was performed 5 times, the highest classification accuracy reached 99.10%.

Figure 7 shows part of the case images of wrong classification. The texture correlation of these misclassified images is not strong, while the ECLBP algorithm has strong robustness to the macro information of the image and has high classification accuracy for the complex texture images with strong texture correlation. Since the feature vector of the proposed algorithm can contain a large amount of macro information, the algorithm has

Table 4 Classification rate (%) on KTHTIPS using different number of train image

Number	10	20	30	40
ECLBP_4,3	69.50	76.80	88.52	95.06
ECLBP_6,5	73.83	80.25	88.89	94.20
ECLBP_8,7	75.06	82.22	88.77	93.09
ECLBP_10,9	74.94	79.88	86.91	92.10
ECLBP_12,11	73.70	77.90	85.68	91.36
ECLBP_4,3|6,5|8,7|10,9|12,11	76.36	83.42	89.43	99.10

a good effect on the image classification with strong texture correlation, and the texture classification accuracy of such images can reach more than 99% (Table 2). However, for images with weak texture correlation, the proposed algorithm in this paper is not able to accurately classify them. For the extended component of the ECLBP algorithm, the extended radius f and filter window size α can be obtained through optimization or grid search. In this paper, parameter 4,3|6,5|8,7|10,9 is recommended to be used for extension, which can achieve classification accuracy of more than 99% and the calculation speed of 0.1 s per image. In the experimental results of ECLBP_12,11, the texture classification was carried out by feature coding with an extended radius of 12 and a filtering window of 11*11. Due to the large extended radius, the information of neighboring center pixels was not included, resulting in the decrease of classification accuracy. The purpose of this experiment is to verify the distance relationship between the center pixel and the extension point, and the experimental results show that when the extension radius is too large, the feature coding will be instability.

5 Conclusion

In this paper, we draw on the theoretical basis of the symbol component and magnitude component of CLBP algorithm, and proposed an extended complete local binary pattern (ECLBP). In the ECLBP algorithm, we also refer to the idea of the Freak algorithm, using region overlap to enhance the correlation between adjacent regions. In order to improve the robustness of the algorithm to image noise, the window grayscale response value of the Gaussian filter and the median filter is used to process individual components instead of a single pixel in the ECLBP algorithm. In the process of classification, the feature information of each “extension circle” of ECLBP algorithm is linked together to form the final feature information. The proposed algorithm ECLBP improved and extended CLBP algorithm, and a completely new concept is introduced. It ensures that the ECLBP algorithm inherits the advantages of the CLBP algorithm and gets further improvement. Because of the introduction of the Freak algorithm model and the concept of median filtering, the ECLBP algorithm can extract more complete feature information than other LBP variant algorithms and has stronger robustness on the impact of noise. Furthermore, the ECLBP algorithm is different from MRELBP, in each layer extension the ECLBP algorithm only need to compute the expansion of the specific component coding, which makes the algorithm have bigger promotion in the expansion flexibility. At the same time, more macro information can be introduced into feature vectors to extract complex texture features on the basis of performance overhead. Since the extended component of the ECLBP algorithm is generated only for its own features, the introduction of noise is avoided, and the final classification accuracy will not be affected even if the extended component of higher level numbers is introduced. Secondly, the time complexity of the ECLBP algorithm is O(n), and the algorithm speed is higher than that of MRLBP algorithm. Finally we designed an experiment to compare the proposed algorithm with LBP, LTP, CLBP and MRELBP in the same experimental condition, and the experimental results show that the proposed algorithm in this paper achieves the best classification effect, and the classification accuracy in the Outex_TC10 and OutexTC_12 data sets has reached more than 99%. In addition, the influence of different filter types and different training sample numbers on the proposed algorithm is also analyzed through experiments. The experimental results show that the median filter is more suitable for the proposed algorithm, but it is not much different from the Gaussian filter. As the training samples increase, the classification accuracy of the proposed algorithm is also greatly improved. ECLBP algorithm also has some shortcomings. ECLBP algorithm expands the traditional model and improves the accuracy of the algorithm, but at the same time, due to the introduction of the expansion model, the dimension of feature coding also increases, which limits the high-speed and real-time performance of LBP algorithm. The next step of this paper will focus on the speed improvement of ECLBP algorithm. PCA and other algorithms can be introduced to reduce the dimension of ECLBP coding, so that ECLBP algorithm can play a role in more scenarios.

Acknowledgements This work was supported in part by Kunming University Introduced Talent Project (XJ20210005), Joint Special Fund Project of Fundamental Research of Local Undergraduate Universities in Yunnan Province, the Key scientific Research Project of Sichuan Science and Technology Department (18ZA0100) and the Young Scholar Leadership Fund of CUIT (J201709).

References

1. Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: application to face recognition. Bahria Univ J Inf Commun Technol 5(1):46–50
2. ChenJ, Kellokumpu V, Zhao G, Pietikainen M. RLBP: Robust Local Binary Pattern. pp 122.1–122.11
3. Csurka G. Visual categorization with bag of keypoints.
4. Fathi A, Naghsh-Nilchi AR (2012) Noise tolerant local binary pattern operator for efficient texture analysis. Pattern Recogn Lett 33(9):1093–1100
5. Guo Z, Zhang L, Zhang D (2010) A completed modeling of local binary pattern operator for texture classification. IEEE Trans Image Process 19(6):1657–1663
6. Guo Z, Zhang L, Zhang D (2010) Rotation invariant texture classification using LBP variance (LBPV) with global matching[J]. Pattern Recogn 43(3):706–719
7. Guo Y, Zhao G, Pietikäinen M (2012) Discriminative features for texture description. Pattern Recogn 45(10):3834–3843
8. Hafiane A, Seetharaman G, Zavidovique B. Median Binary Pattern for Textures Classification. pp 387–398
9. Hayman E, Caputo B, Fritz M, Eklundh JO (2004) On the significance of real-world conditions for material classification. Springer, Berlin Heidelberg
10. Hazgui M, Ghazouani H, Barhoumi W (2021) Genetic programming-based fusion of HOG and LBP features for fully automated texture classification[J]. Vis Comput 37(1)
11. Hegenbart S, Uhl A (2015) A scale- and orientation-adaptive extension of Local Binary Patterns for texture classification[J]. Pattern Recogn 48(8):2633–2644
12. HuangX, Li SZ, Wang Y. Shape localization based on statistical method using extended local binary pattern. pp 184–187
13. Iakovidis DK, Keramidas EG, Maroulis D (2008) Fuzzy Local Binary Patterns for Ultrasound Texture Characterization.
14. Julesz B (1981) Textons, the elements of texture perception, and their interactions. Nature 290(5802):91
15. Kou Q et al (2019) Principal curvatures based local binary pattern for rotation invariant texture classification. Optik Int J Light Electron Opt 193:1629993
16. Leung T, Malik J (2001) Representing and recognizing the visual appearance of materials using three-dimensional textons. Int J Comput Vision 43(1):29–44
17. Leutenegger S, Chli M, Siegwart RY. BRISK: Binary Robust invariant scalable keypoints. pp 2548–2555
18. Li L, Lao S, Fieguth PW, Guo Y, Wang X, Pietikäinen M (2016) Median robust extended local binary pattern for texture classification. IEEE Trans Image Process 25(3):1368–1381
19. Liao S, Law MWK, Chung ACS (2009) Dominant local binary patterns for texture classification. IEEE Trans Image Process A Publ IEEE Signal Process Soc 18(5):1107
20. Liao X, Li K, Zhu X, Liu KJR (2020) Robust Detection of Image Operator Chain with Two-stream Convolutional Neural Network. IEEE J Select Top Signal Process

21. Liao X, Yin J, Chen M, Qin Z (2020) Adaptive Payload Distribution in Multiple Images Steganography Based on Image Texture Features. IEEE Transactions on Dependable and Secure Computing
22. Liao X, Yu Y, Li B, Li Z, Qin Z (2019) A New Payload Partition Strategy in Color Image Steganography. IEEE Transactions on Circuits and Systems for Video Technology, 2019.
23. Liu Li, Lao S, Fieguth PW, Guo Y, Wang X, Pietikainen M (2016) Median robust extended local binary pattern for texture classification. IEEE Trans Image Process 25(3):1368–1381
24. Liu L, Zhao L, Long Y, Kuang G, Fieguth P (2012) Extended local binary patterns for texture classification ☆. Image Vis Comput 30(2):86–99
25. Miao X, Zhao W, Li X et al (2019) Structure descriptor based on just noticeable difference for texture image classification[J]. Appl Opt 58(24):6504
26. Moscoso Thompson E, Biasotti S (2018) Description and retrieval of geometric patterns on surface meshes using an edge-based LBP approach[J]. Pattern Recogn 82:1–15
27. Ojala T, Mäenpää T, Pietikäinen M, Viertola J, Kyllönen J, Huovinen S (2002) Outex - New framework for empirical evaluation of texture analysis algorithms". Proc ICPR 1(1):701–706
28. Ojala T, Pietikainen M, Maenpaa T (2000) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans Pattern Anal Mach Intell 24(7):971–987
29. Ojansivu V, Rahtu E, Heikkila J. Rotation invariant local phase quantization for blur insensitive texture analysis. pp 1–4
30. Pan Z, Wu X, Li Z (2020) Scale-adaptive local binary pattern for texture classification[J]. Multimed Tools Appl 79(4)
31. Qi X, Xiao R, Li CG, Qiao Y, Guo J, Tang X (2014) Pairwise rotation invariant co-occurrence local binary pattern. Pattern Anal Mach Intell IEEE Trans 36(11):2199–2213
32. Ren J, Jiang X, Yuan J (2013) Noise-resistant local binary pattern with an embedded error-correction mechanism. IEEE Trans Image Process 22(10):4049–4060
33. Shakoor MH, Boostani R (2021) Noise robust and rotation invariant texture classification based on local distribution transform. Multimed Tools Appl 80(12):1–28
34. Tan X, Triggs B (2010) Enhanced local texture feature sets for face recognition under difficult lighting conditions. IEEE Trans Image Process 19(6):1635–1650
35. Tekin R, Erturul MF, Kaya Y (2020) New local binary pattern approaches based on color channels in texture classification. Multimed Tools Appl
36. Vandergheynst P, Ortiz R, Alahi A. FREAK: Fast Retina Keypoint. pp 510–517
37. Varma M, Zisserman A (2005) A statistical approach to texture classification from single images. Int J Comput Vision 62(1–2):61–81
38. Varma M, Zisserman A (2009) A statistical approach to material classification using image patch exemplars. IEEE Trans Pattern Anal Mach Intell 31(11):2032–2047
39. Zhang J, Tan T (2002) Brief review of invariant texture analysis methods ☆. Pattern Recogn 35(3):735–747
40. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans Pattern Anal Mach Intell 29(6):915–928
41. Zhou LF, Du YW, Li WS et al (2018) Pose-robust face recognition with huffman-lbp enhanced by divide-and-rule strategy[J]. Pattern Recogn 78:43–55