Multi-Scale Local Spatial Binary Patterns for Content-Based Image Retrieval

Yu Xia, Shouhong Wan, Peiquan Jin, and Lihua Yue

School of Computer Science and Technology
University of Science and Technology of China, China
xiay1989@mail.ustc.edu.cn,{wansh,jpq,llyue}@ustc.edu.cn

Abstract. Content-based image retrieval (CBIR) has been widely studied in recent years. CBIR usually employs feature descriptors to describe the concerned characters of images, such as geometric descriptor and texture descriptor. Many texture descriptors in texture analysis and image retrieval are based on the so-called Local Binary Pattern (LBP) technique. However, LBP lacks of the spatial distribution information of texture features. In this paper, we aim at improving the traditional LBP and present a novel texture feature descriptor for CBIR called Multi-Scale Local Spatial Binary Patterns (MLSBP). MLSBP integrates LBP with spatial distribution information of gray-level variation direction and gray-level variation between the referenced pixel and its neighbors. In addition, MLSBP extracts the texture features from images on different scale levels. We conduct experiments to compare the performance of MLSBP with five competitors including LBP, Uniform LBP (ULBP), Completed LBP (CLBP), Local Ternary Patterns (LTP), and Local Tetra Patterns (LTrP). Also three benchmark image databases are used in the measurement, which are the Bradotz Texture Database (DB1), the MIT VisTex Database (DB2), and the Corel 1000 Database (DB3). The experimental results show that MLSBP is superior to the competitive algorithms in terms of precision and recall.

Keywords: Content-based image retrieval, Local binary pattern, Texture feature, Spatial distribution.

1 Introduction

Image retrieval is one of the important applications of image processing [1]. Due to the explosive growth of digital images, there exists an urgent need for efficient image retrieval algorithms which can search the desired images from databases. Generally, there are three main categories of image retrieval: text-based image retrieval (TBIR), content-based image retrieval (CBIR) and semantic-based image retrieval (SBIR) [2]. Images in TBIR systems need to be manually annotated and require much human labor, and the annotation accuracy is subject to human perception [3]. For SBIR, researchers focus on the representation of high-level semantic of images, while low-level features (color, texture, shape, etc.) often fail to describe the high-level semantic concepts of images [4]. Content-based image retrieval based on the visual contents of

T. Yoshida et al. (Eds.): AMT 2013, LNCS 8210, pp. 423–432, 2013.

an image, such as color, texture, shape, distribution layout, etc., to represent the images. Due to the advantages of simple calculation and high efficiency, CBIR is becoming more widely used in image retrieval. The difficulty in CBIR is to find a best representation of an image for all perceptual subjectivity, due to the different view angles and illumination changes.

In this paper, we aim at improving the traditional LBP and present a novel texture feature descriptor for CBIR, which is called Multi-Scale Local Spatial Binary Patterns (MLSBP). MLSBP integrates LBP with spatial distribution information of gray-level variation direction and gray-level variation between the referenced pixel and its neighbors. In addition, MLSBP extracts the texture features from images on different scale levels. Finally, we conduct three experiments to demonstrate the performance of MLSBP and other compared methods.

The remainder of the paper is structured as follows. Section 2 introduces the related work. Section 3 gives the calculation of MLSBP. Section 4 presents the framework of proposed image retrieval algorithm and the similarity measurement. Section 5 shows our experiment results. Section 6 concludes our work and gives the directions for future studies.

2 Related Work

Due to the efficiency to image processing, texture features have been widely used in CBIR. LBP [5] has emerged as an efficient feature in texture analysis and CBIR. With the advantages of simple calculation and multi-scale characteristic, LBP performs excellent in texture image classification. Recent years, LBP has been promoted to different versions: uniform LBP (ULBP) [6], completed LBP (LBP) [7] and local ternary patterns (LTP) [8]. LBP extracted the gray-level variation pattern between center pixel and its surrounding neighbors. ULBP based on the fact that most frequent 'uniform' binary patterns correspond to primitive micro-features, such as edges, corners, and spots. Thus, ULBP reduces LBP patterns into fewer uniform patterns. At the same time, by choosing the minimum of pattern values after circular right shifting, ULBP obtained the characteristic of rotational invariance. CLBP completed LBP with magnitude information (CLBP_M) and the gray level represented by center pixels (CLBP_C). By combing CLBP_M, CLBP_C with the traditional LBP, CLBP can extract texture feature more comprehensively. LTP extended LBP with coding variation pattern by ternary result and respectively calculated the upper and lower value as the features. The local tetra patterns (LTrP) [9] based on the first-order derivatives in vertical and horizontal directions and encoded the variation with tetra codes. And through calculating high-order patterns, LTrP can extract more detail texture information. All these patterns only calculated the gray variation of pixels, but ignored the spatial distribution information of variation and direction. Thus, it is evident that the performance of these methods can be improved by extracting spatial distribution information.

3 Multi-Scale Local Spatial Binary Patterns (MLSBP)

3.1 Local Patterns

LBP was introduced in [5] for texture classification. Given a referenced center pixel, LBP is computed by comparing its gray value with its neighbors, based on Formula 3.1.

$$LBP_{P,R} = \sum_{p=1}^{P} 2^{(p-1)} \times f_{LBP}(g_p - g_c)$$
$$f_{LBP} = \begin{cases} 1, & x \geq 0 \\ 0, & else \end{cases} \tag{3.1}$$

where g_c is the gray-level value of the referenced pixel, g_p is the gray-level value of its neighbors, P is the number of neighbors, and R is the radius of the neighbors.

LTP was introduced in [8] for face recognition. LTP extended LBP to a three-valued code, and calculated the upper and lower value. LTP is computed by Formula 3.2.

$$LTP^{upper} = \sum_{p=1}^{P} 2^{(p-1)}, if \quad f_{LTP}(g_p - g_c) = 1$$
$$LTP^{lower} = \sum_{p=1}^{P} 2^{(p-1)}, if \quad f_{LTP}(g_p - g_c) = -1 \tag{3.2}$$
$$f_{LTP}(x, y, t) = \begin{cases} 1, & x \geq y + t \\ 0, & |x - y| < t \\ -1, & x \leq y - t \end{cases}$$

where t is the threshold. More details about LTP can be found in [8].

3.2 Local Spatial Binary Patterns (LSBP)

LSBP we proposed extracts the spatial distribution information of gray-level variation, which is an effective supplement to LBP. LSBP is defined as Formula 3.3.

$$LSBP^{\alpha}(i, j) = (GVP_P^{\alpha} = i, GVP_Q^{\alpha} = j) \tag{3.3}$$

where P and Q are two pixels which are the distance **d** apart in original image, GVP_P^{α} and GVP_Q^{α} are the gray-level variation patterns (*GVP*) of pixel P and Q on direction α.

The *GVP* can be obtained by integrating the relationship results together on each direction. The relationship $f_{relation}(P,Q)$ (binary coding) between two gray values P and Q can be calculated by Formula 3.4.

$$f_{relation}(P,Q)=\begin{cases}00, & if\ (P-Q)\geq T\ \&\ P\geq Q\\ 01, & if\ (P-Q)<T\ \&\ P\geq Q\\ 10, & if\ (Q-P)\geq T\ \&\ P<Q\\ 11, & if\ (Q-P)\geq T\ \&\ P<Q\end{cases} \tag{3.4}$$

where T is chosen as a threshold to distinguish two pixels' gray-level values. Supposing T is small, gray-level variations are coded more precisely, which means variations with small difference can be coded as different results. This will lead to the inaccuracy of similarity measurement. On the contrary, supposing T is large, variations are coded more roughly, which means variations with big difference can be coded as the same results. This will lead to the loss of texture detail information.

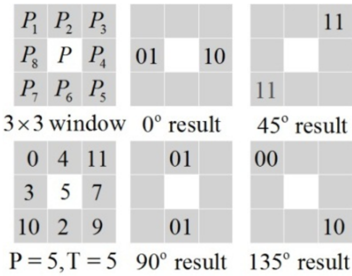


Fig. 1. Calculation of gray-level variation pattern

Fig. 1 shows an example of 9 pixels in 3×3 window, considering the center referenced pixel is P and its surrounding 8 neighbors are Q, with T is selected as 5. Based on the code results of each pixel, we can get a four binary code by integrating the two neighbors' results together on each direction to get gray-level variation pattern (*GVP*). The neighbors of P on 45° are P_3 and P_7, thus, *GVP* on 45° in the example can be coded as "1111", while 0° is "1001", 90° is "0101", 135° is "0010". We can get one pattern image on each direction with every pixel value in pattern image ranging from "0000" to "1111". Finally, LSBP value can be obtained based on the *GVP* results. Each entry (i, j) in LSBP corresponds to the number of occurrences of the pair of i and j in *GVP* image. To simplify the computation, LSBP (i, j) can be calculated by integrating the *GVP* of i and j together. For example, supposing *GVP* of i is "1101", j is "0110", then LSBP (i, j) can be coded as "11010110", which can be converted to decimal number "214". Through recording the occurrences of different LSBP codes (from "00000000" (0) to "11111111" (255)), LSBP histograms can be constructed to represent the spatial distribution information of texture in images. In this paper, we choose **d** as (1, 0), (0, 1), (1, 1), (-1, 1), which means the LSBP is respectively calculated with four pairs. When **d** is set as (1, 0), e.g., the pair of pixels is P and P_4. Based on the four *GVP* pattern images, we can obtain sixteen LSBP histograms, every four LSBP histograms on each direction.

Furthermore, we calculate the mean and standard deviation of the variations between pixels and its surrounding neighbors to reflect the gray-level variation magnitude information. Assuming the center pixel is P and the whole three pixels' values on direction α are (P_x, P, P_y). The variation Var_P^{α} is calculated by Formula 3.5.

$$Var_P^{\alpha} = |P_x - P| + |P - P_y| \quad (3.5)$$

For the pixels whose pattern values on direction α are m, the mean ($\overline{V}_m^{\alpha}$) and standard deviation (σ_m^{α}) are calculated by Formula 3.6.

$$\overline{V}_m^{\alpha} = \sum_{i=1}^{K} Var_{P_i}^{\alpha} / K, P_i \in \Phi_{MP}, \sigma_m^{\alpha} = \sqrt{\sum_{i=1}^{K} (Var_{P_i}^{\alpha} - \overline{V}_m^{\alpha})^2 / K}, P_i \in \Phi_{MP} \quad (3.6)$$

where Φ_{MP} is the pixels' set whose *GVP* value is m on direction α, K is the number of Φ_{MP}. Through this step, we can calculate the mean and standard deviation of the 16 different patterns on 0°, 45°, 90°, and 135° directions, and a $4 \times 16 \times 2 = 128$ dimensional variation magnitude vector (**VM**) was obtained to represent the gray-level variation magnitude information.

3.3 Calculation of Multi-Scale LSBP

In order to extract more detail information and contour information, multi-scale analysis is always been used in texture analysis. Due to the different combinations of P and R in Formula.(1), the LBP has the superiority of multi-scale characteristics. As Fig. 2 shows, firstly, we down-sampling the original image with 3×3 and 5×5 window. Secondly, the LSBP vectors of the sampling images were extracted. The final step is normalization of LSBP vectors to construct the MLSBP feature vector.

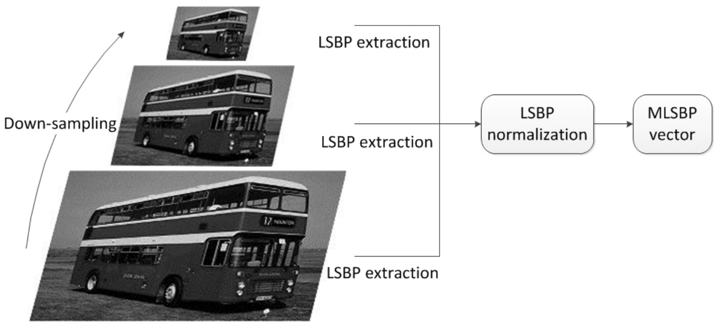


Fig. 2. Feature extraction of Multi-scale LSBP

Given an image *G*, it can be sampled by Formula 3.7.

$$G'(x, y) = \frac{1}{w * w} \sum_{i=-(w-1)/2}^{(w-1)/2} \sum_{j=-(w-1)/2}^{(w-1)/2} G(x+i, y+j) \quad (3.7)$$

where $w*w$ is the size of sampling window and $G(x, y)$ is one of the pixels in G. In this paper, we respectively choose w as 3 and 5. After down-sampling, the height and width of G' is reduced to $1/w$ of the height and width of G. Suppose the texture feature vectors of original image, 3×3 sampling image and 5×5 sampling image are f_1^{LSBP}, f_3^{LSBP} and f_5^{LSBP}. The MLSBP vectors can be defined by Formula 3.8.

$$f^{MLSBP} = (f_1^{LSBP}, f_3^{LSBP}, f_5^{LSBP}) \tag{3.8}$$

After the MLSBP extraction, we can use f^{MLSBP} as the feature vectors to measure the similarity between query image and images from database.

4 The Image Retrieval Algorithm Using MLSBP

4.1 Proposed Image Retrieval Framework

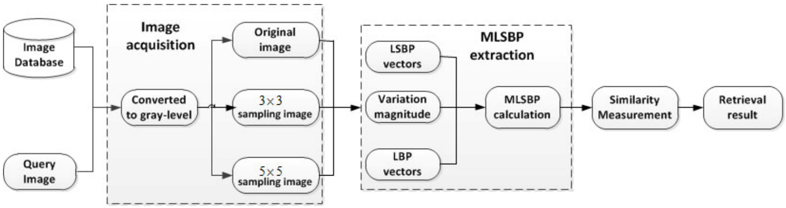


Fig. 3. The framework of proposed image retrieval algorithm

The framework of the proposed image retrieval algorithm is shown in Fig. 3. Firstly, the query image and the images in database are converted into gray images. Then we filter the original image with 3×3 and 5×5 windows. Secondly, LSBP feature vectors of 0°, 45°, 90° and 135°, variation magnitude vectors and LBP feature vectors are extracted from the original and filtered images. After the step of calculating the MLSBP vectors, we can obtain the MLSBP extraction results. Finally, we use feature similarity to measure the similarity between the query image and the database images, and return retrieval result based on similarity calculation results.

4.2 Feature Similarity Measurement

Supposing the feature vector of the query image (Q) and images in database (DB) are represented as $f_Q(f_{Q_1}, f_{Q_2}, ..., f_{Q_L})$ and $f_{DB}(f_{DB_1}, f_{DB_2}, ..., f_{DB_L})$. L is the dimension of feature vector. This involves the selection of n top-matched images by measuring the distance between the query image and the images in database. The similarity distance to match the images is computed using Formula 4.1.

$$d = \sum_{i=1}^{L} \left| \frac{f_{DB_i} - f_{Q_i}}{1 + f_{DB_i} + f_{Q_i}} \right| \tag{4.1}$$

where f_{Q_i} and f_{DB_i} is the *ith* feature value of the feature vector. As we can know, the smaller d is, the more similar two pictures are.

5 Experiment Results

5.1 Experiment Setup

In order to prove the effectiveness of MLSBP, experiments are implemented on three benchmark databases: Bradotz Texture Database, MIT VisTex Database and CoreImage 1000 Database. The Bradotz texture database and MIT VisTex database consist of texture images, while the CoreImage 1000 database consists of natural scene images. The image retrieval algorithm we proposed is implemented by C++ and OPENCV. The performance of the proposed image retrieval algorithm is measured in terms of precision, recall, average precision (AP), and average recall (AR). For the query image *Q*, the indicators are defined as Formula 5.1 to 5.4.

$$\Pr ecision(I_q,n)=\frac{1}{n}\sum_{i=1}^{|DB|}\left|\delta(\phi(I_i),\phi(I_q))\,|\,Rank(I_i,I_q)\le n\right| \tag{5.1}$$

$$\operatorname{Re}call(I_q,n)=\frac{1}{N_G}\sum_{i=1}^{|DB|}\left|\delta(\phi(I_i),\phi(I_q))\,|\,Rank(I_i,I_q)\le n\right| \tag{5.2}$$

$$AP(n)=\frac{1}{|DB|}\sum_{i=1}^{|DB|}\Pr ecision(I_i,n),\ AR(n)=\frac{1}{|DB|}\sum_{i=1}^{|DB|}\operatorname{Re}call(I_i,n) \tag{5.3}$$

$$\delta(\phi(I_i),\phi(I_q))=\begin{cases}1,\phi(I_i)=\phi(I_q)\\ 0,else\end{cases} \tag{5.4}$$

where n indicates the number of retrieved images, and $|DB|$ is the size of the image database. $\phi(x)$ is the category of image x, $Rank(I_i,I_q)$ return the rank of image I_i (for the query image I_q) among all images of $|DB|$, N_G is the size of each image category in database.Given in Table.1 are the abbreviations of different methods used in the experimental discussions. In our experiments, the *T* used in LTP is set as 10.

Table 1. The abbreviations of different methods

Abbreviations	Methods
MLSBP	Multi-scale local spatial binary patterns
LSBP	Local spatial binary patterns
LBP	Local binary patterns
ULBP	Uniform binary patterns
CLBP	Completed local binary patterns
LTP	Local ternary patterns
LTrP	Local tetra patterns

5.2 Experiment on DB1

Database DB1 is consists of 111 different textures and used in experiment 1. These 111 textures are from Brodatz texture photographic album [10]. The size of each texture is 512×512. Each 512×512 image is divided into sixteen non-overlapping sub-images, thus creating a database of 1776 (16×111) images. Each image in the database is considered as the query image. Examples of DB1 are shown in Fig. 4. Fig. 5 illustrates the retrieval performance of the proposed method and other existing methods. T is the threshold chosen in Formula. (4). In this experiment, we set T as 10. As Fig. 5 shows, MLSBP outperforms the other existing methods.

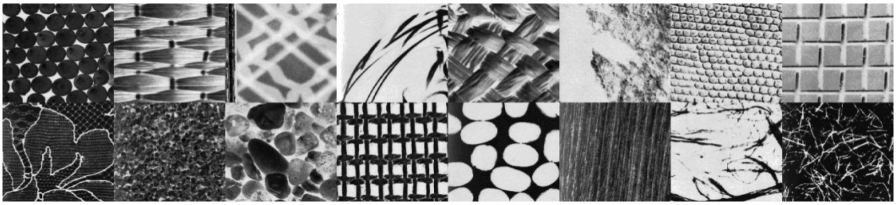
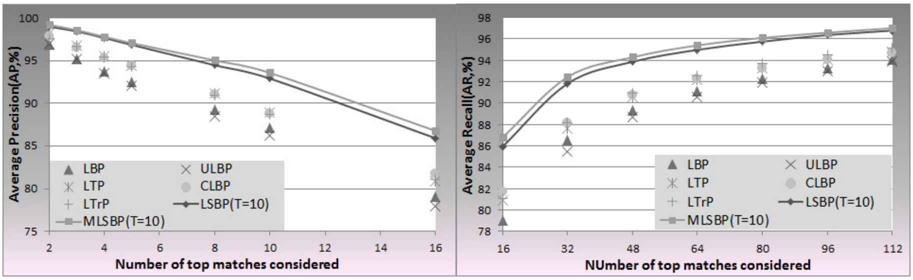
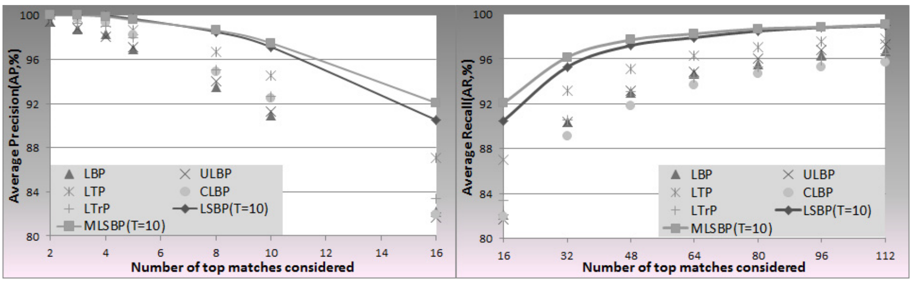


Fig. 4. Examples of texture images in DB1



(a) AP results of existing methods on DB1 (b) AR results of existing methods on DB1

Fig. 5. Comparison of the MLSBP with other existing methods on DB1



(a) AP results of existing methods on DB2 (b) AR results of existing methods on DB2

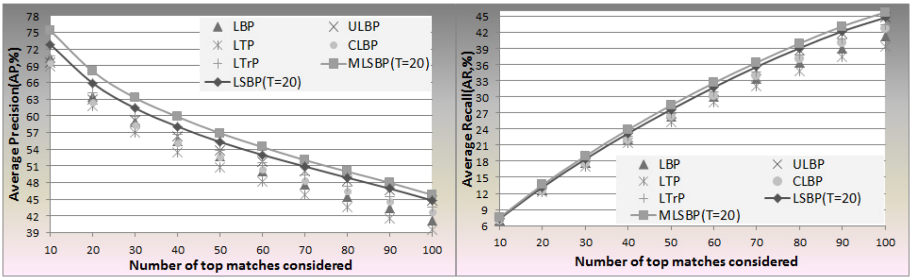
Fig. 6. Comparison of the MLSBP with other existing methods on DB2

5.3 Experiment on DB2

In this experiment, database DB2 is consists of 40 different textures collected from the MIT VisTex database [12]. The size of each texture image is 512×512, each image is divided into sixteen 128×128 non-overlapping sub-images, creating a database of 640 (16×40) images. In this experiment, we set T as 10. The retrieval result of DB2 is shown in Fig. 6.

5.4 Experiment on DB3

In experiment 3, images from the Corel database [11] have been used. This database consists of a large number of images of various contents ranging from animal images to outdoor sports and natural images. These images have been pre-classified into different categories. Each category has 100 images. For our experiment, we have chosen 1000 images to form DB3. These images are collected from ten different domains: Africans, Beach, Buildings, Buses, Dinosaurs, Elephants, Flowers, Horses, Mountains, Food. In this experiment, we set T as 20. As Fig. 7 shows, MLSBP performs better on average precision rate and average recall rate than other existing methods on DB3.



(a) AP results of existing methods on DB3 (b) AR results of existing methods on DB3

Fig. 7. Comparison of the MLSBP with other existing methods on DB3

The accurate results of AP and AR are shown in Table 2. It is evident that MLSBP outperforms other existing methods, respectively in texture images and natural scene images.

Table 2. The AP and AR results of all databases (The value in brackets is n in Formula.(12))

Methods	Average Precision(%)			Average Recall(%)		
	DB1(16)	DB2(16)	DB3(10)	DB1(112)	DB2(112)	DB3(100)
MLSBP	**86.78**	**92.06**	**75.39**	**97.00**	**99.04**	**45.81**
LSBP	85.92	90.50	72.88	96.78	98.97	44.81
LBP	79.01	82.13	70.49	94.07	96.70	41.30
ULBP	77.99	81.67	69.51	93.86	97.31	44.81
CLBP	80.86	87.05	68.90	94.78	97.80	39.48
LTP	81.72	81.92	69.55	94.70	95.73	42.71
LTrP	81.10	83.36	70.78	95.13	96.46	43.55

6 Conclusion

In this paper, we present a new texture feature descriptor referred to as MLSBP for CBIR. MLSBP not only extract the spatial distribution information of gray-level variation, but also extract the multi-scale texture feature from different scales of original image. However, MLSBP only depends on the gray image, but lose the color information of natural scene image. So the retrieval results of natural images are not every good. At the same time, the dimension of MLSBP is very large, which will influence the retrieval efficiency. In the future, we will focus our research on how to reduce the dimension of MLSBP vectors and combine MLSBP with color features to improve the retrieval accuracy.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Grant No. 61272317) and the General Program of Natural Science Foundation of AnHui of China (Grant No. 1208085MF90).

References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Computing Surveys (CSUR) 40(2), 5 (2008)
2. Liu, G.-H., Zhang, L., Hou, Y.-K., Li, Z.-Y., Yang, J.-Y.: Image retrieval based on multi-texton histogram. Pattern Recognition 43(7), 2380–2389 (2010)
3. Li, W., Duan, L., Xu, D., Tsang, I.W.-H.: Text-based image retrieval using progressive multi-instance learning. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2049–2055. IEEE (2011)
4. Liu, Y., Zhang, D., Lu, G., Ma, W.-Y.: A survey of content-based image retrieval with high-level semantics. Pattern Recognition 40(1), 262–282 (2007)
5. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognition 29(1), 51–59 (1996)
6. 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 24(7), 971–987 (2002)
7. Guo, Z., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Transactions on Image Processing 19(6), 1657–1663 (2010)
8. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. In: Zhou, S.K., Zhao, W., Tang, X., Gong, S. (eds.) AMFG 2007. LNCS, vol. 4778, pp. 168–182. Springer, Heidelberg (2007)
9. Murala, S., Maheshwari, R.P., Balasubramanian, R.: Local tetra patterns: a new feature descriptor for content-based image retrieval. IEEE Transactions on Image Processing 21(5), 2874–2886 (2012)
10. Brodatz, P.: Textures: A Photographic Album for Artists and Designers. Dover, New York (1996)
11. Corel 10000 image database, `http://wang.ist.psu.edu/docs/related.shtml`
12. MIT Vision and Modeling Group, Cambridge, Vision texture, `http://vismod.media.mit.edu/pub/`