Lacunarity Analysis on Image Patterns for Texture Classification*

Yuhui Quan[1,2], Yong Xu[1], Yuping Sun[1,2] and Yu Luo[1,2]
[1]School of Computer Science & Engineering, South China University of Technology, Guangzhou 510006, China
[2]Department of Mathematics, National University of Singapore, Singapore 117542

Abstract

Based on the concept of lacunarity in fractal geometry, we developed a statistical approach to texture description, which yields highly discriminative feature with strong robustness to a wide range of transformations, including photometric changes and geometric changes. The texture feature is constructed by concatenating the lacunarity-related parameters estimated from the multi-scale local binary patterns of image. Benefiting from the ability of lacunarity analysis to distinguish spatial patterns, our method is able to characterize the spatial distribution of local image structures from multiple scales. The proposed feature was applied to texture classification and has demonstrated excellent performance in comparison with several state-of-the-art approaches on four benchmark datasets.

1. Introduction

Texture is a fundamental part of visual feature, since images often exhibit variations of intensities with certain repeated patterns. Texture provides a powerful cue for many vision-related applications, such as material classification, object recognition, natural scene identification. Although it is easy for human to identify texture, defining texture is challenging. Many existing texture description and classification methods (e.g., [11, 18, 12, 34, 32]) model texture as a collage collected from certain types of textons. As a result, texture is represented as histogram of local image patterns.

The representation of local image patterns differs in the existing methods: it can be predefined, such as using filter response [29, 6, 10], binary codes [23, 16], and templates [28]; or be adaptive to image, e.g., using SIFT features [15, 32], affine-invariant regions [13], random projections of image patches [14], and local feature clusters [27]. The adaptive image pattern representation usually involves feature detection and clustering technique. Thus the resultant texture descriptors often show strong robustness to geometric and illumination changes, as well as partial occlusions. However, the extracted patterns are sparse over image space, implying that the resultant features inevitably lose details in discrimination. Moreover, the clustering process makes the computational cost dramatically increase.

In contrast to the adaptive ones, the predefined image patterns are more discriminative but with weaker invariance. This inspires us to develop a robust statistical method to integrate such image patterns into a global feature that enjoys both robustness and discriminability. Our work is motivated by the observation that the spatial distribution of local image patterns exhibits statistical self-similarities within a certain range of scales [10]. Such self-similarities can be well described by the so-called fractal geometry.

In recent years, fractal analysis has emerged as a promising approach to capturing the self-similarities of texture. Based on fractal analysis, many successful approaches (e.g., [10, 29, 28]) have been proposed for texture classification. The basic idea of these methods is to use multiple fractal dimensions to summarize the spatial distribution of image patterns. While the fractal-dimension-based analysis used in these methods has led to impressive results, other powerful tools for fractal analysis have not been fully exploited, and one of them is the so-called lacunarity analysis. Compared with fractal dimension, lacunarity is more general in characterizing spatial features and can be readily used to describe multi-fractal and even non-fractal patterns [24]. In the past, several lacunarity-analysis-based methods (e.g., [25, 20, 17]) have been proposed. However, these methods have not been successfully applied to classifying complex textures from real world.

In this paper, we proposed an effective method to characterize the spatial distribution of image patterns using lacunarity analysis. The extracted features encode the scaling behaviors of the lacunarity of image patterns. Our method was applied to texture classification and evaluated on four benchmark datasets. Our method has demonstrated excellent performance in comparison with the existing approaches. The rest of this paper is organized as follows. Sec. 2 introduces the background knowledge about local bi-

*Project supported by National Nature Science Foundations of China (61273255,61211130308 and 61070091), Fundamental Research Funds for the Central Universities (SCUT 2013ZG0011) and GuangDong Technological innovation project (2013KJCX0010).

nary patterns and lacunarity analysis. Sec. 3 is devoted to the proposed method. The evaluation is reported in Sec. 4 and the conclusion is drawn in Sec. 5.

2. Preliminaries

2.1. Local binary pattern

There is an abundant literature on extracting image patterns. One representative way is the so-called *local binary pattern (LBP)*. The original LBP operator proposed by Ojala *et al*. [21] forms labels for image pixels by thresholding the 3×3 neighborhood of each pixel with the center value and summing the resultant binary numbers weighted by powers of two. To adapt the LBP operator to the neighborhoods of different sizes [23], a circular symmetric neighborhood denoted by (P, R) is defined. Here P denotes the number of the sampling points and R denotes the radius of the neighborhood. The pixel value of a sampling point is bilinearly interpolated if the point does not lie at the integer coordinates. The modified operator, denoted by $\text{LBP}_{P,R}$, can be written as

$$\text{LBP}_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) * 2^p, \tag{1}$$

where $s(x)$ is the thresholding function that is assigned the value 1 if x is positive and 0 if negative, g_c is the gray value of the center pixel and g_p $(p = 0, 1, ..., P-1)$ is the gray value of the neighbors. Since $s(g_p - g_c)$ is invariant to any monotonic photometric changes, the operator $\text{LBP}_{P,R}$ is robust to lighting changes.

However, the operator $\text{LBP}_{P,R}$ is sensitive to image rotation. Hence, the rotation-invariant LBP operator [22], denoted by $\text{LBP}^{ri}_{P,R}$, is developed by circularly rotating each LBP binary code into its minimum value:

$$\text{LBP}^{ri}_{P,R} = min\{\mathcal{R}(p, \text{LBP}_{P,R}) \mid p = 0, 1, ..., P-1\}, \tag{2}$$

where $\mathcal{R}(p, x)$ performs a circular bit-wise right shift on x by p times. For instance, the bit sequences 10110000, 00101100 and 11000010 arise from different rotations of the same local pattern and they all correspond to the normalized sequence 00001011.

One further extension of LBP is to eliminate the patterns with frequent bitwise jumps in their binary codes, which can reduce the sensitivity to noise. The jump frequency is measured by a uniformity measure U defined as

$$\begin{aligned} U(\text{LBP}_{P,R}) =& |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \\ & \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)|. \end{aligned} \tag{3}$$

The measure U counts the number of bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. A local binary pattern is called uniform [23] if the uniformity measure on the pattern is at most two. The corresponding uniform rotation-invariant LBP operator $\text{LBP}^{riu2}_{P,R}$ is defined as

$$\text{LBP}^{riu2}_{P,R} = \begin{cases} \text{LBP}^{ri}_{P,R}, & if\ U(\text{LBP}_{P,R}) \leq 2; \\ P+1, & otherwise. \end{cases} \tag{4}$$

The operator $\text{LBP}^{riu2}_{P,R}$ assigns a single label to all the non-uniform patterns, which benefits reducing the length of LBP-based feature and implementing a simple rotation-invariant descriptor. By using a look-up table, the calculation of $\text{LBP}^{riu2}_{P,R}$ is very efficient. Note that there are many other LBP variants (e.g., [8, 33]) that have demonstrated better performance in texture description. We employ the LBP operator $\text{LBP}^{riu2}_{P,R}$ for its simplicity. Our results show that such a simple coding strategy can perform well.

2.2. Lacunarity analysis

Lacunarity, originally introduced by Mandelbrot [19], is a specialized term in fractal geometry referring to a measure on how patterns fill space. Geometric objects appear more lacunar if they contain a wide range of gap sizes. More precisely, lacunarity measures the deviation of a geometric object from translational invariance [7]. At a given scale, low lacunarity indicates being homogeneous and transitionally invariant because all gap sizes are the same, whereas objects of high lacunarity are heterogeneous and not transitionally invariant. But note that high-lacunarity objects which are heterogeneous at small scales can be quite homogeneous at larger scales or vice versa. In other words, lacunarity is a scale-dependent measure on the spatial complexity of patterns.

A simple way to calculate lacunarity on a binary image B is the gliding box method [5]. As depicted in Fig. 1, a box of size $r \times r$ is first gliding through the image. The number of the mass points (black pixels) within the box at each position is calculated. A histogram, denoted as $X^B_r(n)$ is then built upon the collection of the values from all the boxes. Here n denotes the number of mass points falling into the box, and $X^B_r(n)$ is the number of the boxes containing n mass points. See Fig. 1 for an illustration of the calculation process. The lacunarity at scale r is defined as

$$\Lambda_r(B) = \frac{E[(X^B_r)^2]}{(E[X^B_r])^2}. \tag{5}$$

The lacunarity $\Lambda_r(B)$ is a scale-dependent variable. For the objects with self-similarities, the lacunarity exhibits power-law behaviors [19] with respect to its scale, i.e.,

$$\Lambda_r(B) \propto (\frac{1}{r})^{D(B)}, \tag{6}$$

where $D(B)$ is a scale-independent exponent.

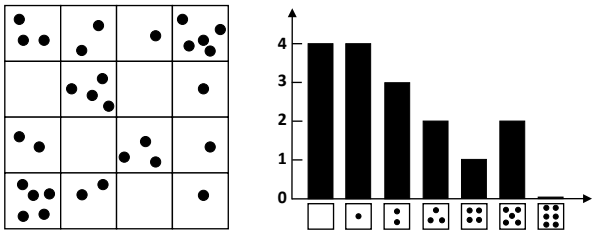


Figure 1. The gliding box travels over the whole image, and is centered on each pixel (this is simplified in the left figure). The number of holes inside the box are calculated. These numbers are used to build a histogram, as depicted in the right figure.

3. Our Method

Since image patterns exhibit power-law behaviors, as discussed in Sec.1, the lacunarity on the image patterns is assumed to satisfy Eqn. (6). By taking logarithm on both sides of Eqn. (6), we obtain

$$\ln \Lambda_r(B) = D(B) \ln r + L(B). \tag{7}$$

The variables $D(B)$ and $L(B)$ indeed encode the scaling behaviors of the lacunarity of image patterns. Hence, for binary image B we derive our lacunarity-related feature denoted by $\text{LAC}(B)$ as follows:

$$\text{LAC}(B) = [D(B), L(B)]. \tag{8}$$

To estimate $D(B)$ and $L(B)$, the linear least square fitting technique is used.

The reliability of the LAC feature depends on the types of the image patterns used in calculation. Here we employ the *local binary patterns* (*LBPs*) presented in Sec 2.1. The reason is the LBPs defined in Eqn. (4) are robust to lighting changes, image rotation and moderate amount of noise. Moreover, the LBPs have computational simplicity and efficiency. To exploit structures existing in different scales, the LBPs are extracted in a multi-scale manner. Given an image I, we compute a sequence of LBP code maps $J_1, J_2, \ldots, J_N$ by applying the uniform rotation-invariant LBP operator $\text{LBP}^{riu2}_{P,R}$ defined in Eqn (4) to I with a series of parameters

$$\{(P_i, R_i), i = 1, 2..., N\}.$$

In details, the code map J_i is a label image generated by applying $\text{LBP}^{riu2}_{P_i,R_i}$ to I. The parameter P_i defines the shape of neighborhood, and R_i determines the size of the neighborhood as well as the scale of the encoded patterns. By using different values of P, the generated code maps can capture various types of local structures in texture, and, by using multiple values of R, the multi-scale analysis is conducted. Note that for digital image, larger R would result in more freedom in choosing P. Our experiment show that, only a few values of R and P are able to characterize the rich structures of texture and achieve excellent performance.

Algorithm 1 Pattern Lacunarity Spectrum (PLS)

Input: Texture image I

Output: Texture feature $\text{PLS}(I)$

1. Calculate LBP code maps $J_1, J_2, ...J_N$ using (1)-(4) with a series of parameters $\{(P_i, R_i), i = 1, ..., N\}$:

$$J_i = \text{LBP}^{riu2}_{P_i,R_i}(I), i = 1, ..., N.$$

2. Generate binary images $\{B_{i,j}, j = 1, ..., P_i + 2\}$ using (9) from each LBP code map:

$$B_{i,j}(x, y) = \begin{cases} 1, & if\ J_i(x, y) = j; \\ 0, & otherwise. \end{cases}$$

3. Compute lacunarity-related features $L_{i,j}$ on each binary image using (7) and (8):

$$L_{i,j} = \text{LAC}(B_{i,j}), i = 1, ..., N; j = 1, ..., P_i + 2.$$

4. Output PLS feature via concatenation over $L_{i,j}$:

$$\text{PLS}(I) = \biguplus_{i,j} L_{i,j}, i = 1, ..., N; j = 1, ..., P_i + 2.$$

Next, a series of binary images $\{B_{i,j}, j = 1, ..., P_i + 2\}$ are generated from each code map J_i via pixel classification with respect to the code value of each LBP:

$$B_{i,j}(x, y) = \begin{cases} 1, & if\ J_i(x, y) = j; \\ 0, & otherwise. \end{cases} \tag{9}$$

See Fig. 3 for some examples of the binary images. Each binary image provides the spatial locations of the image patterns of the same type. To characterize the spatial distribution of the image patterns, the LAC feature is computed using Eqn. 8 on each binary image and concatenated as the *pattern lacunarity spectrum* (PLS) feature:

$$\text{PLS}(I) = \biguplus_{i,j} \text{LAC}(B_{i,j}), i = 1, ..., N, j = 1, ..., P_i + 2, \tag{10}$$

where $\biguplus$ denotes the concatenation of the LAC vectors.

The proposed approach is illustrated in Fig. 2 and outlined in Alg. 1. See Fig. 4 for the illustration of the PLS feature. It can be seen that the PLS feature can enjoy both the inter-class discrimination and intra-class similarity. Note that the number of binary images generated from each LBP code map J_i varies with the parameter (P_i, R_i) of the LBP operator $\text{LBP}^{riu2}_{P_i,R_i}$. By the definition of $\text{LBP}^{riu2}_{P,R}$, the total

length of the proposed PLS feature is $\sum_{i=1}^{N} 2(P_i + 2)$.

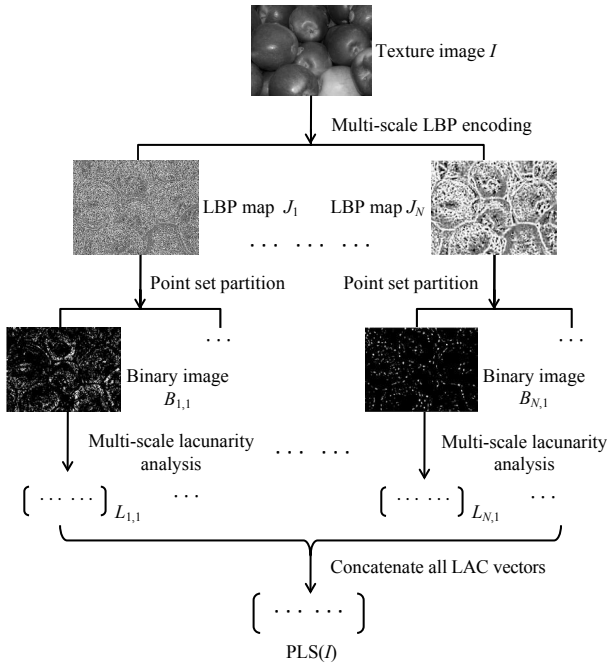


Figure 2. Flowchart of the proposed method.

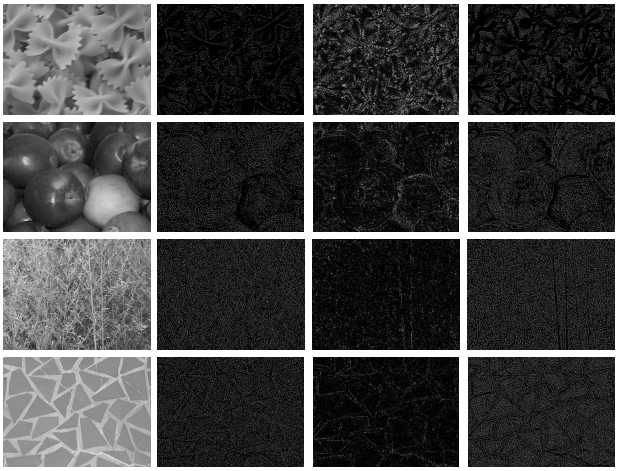


Figure 3. Binary images generated from the LBPs. Column (a) are four different texture images. Column (b)-(d) are the corresponding binary images generated by pixel classification from the LBPs of the original texture images.

4. Experimental Evaluation

In this section, the proposed method is evaluated by applying it to texture classification. The parameters of the proposed method are set as the same through all experiments. The scale range used for estimating lacunarity is set to be a series of integers from 2 to 14. The parameter series $\{(P_i, R_i)\}$ for multi-scale LBP coding are set as $\{(4, 1), (16, 2), (16, 3), (8, 5), (16, 5)\}$. There are two reasons for selecting such LBP parameters. Firstly, the number of the sampling points in each scale is set to be 4, 8 or 16 respectively, according to the corresponding scale. Secondly, for computing the multi-scale LBPs, the scales are defined as a series of integers which start from 1 and is increased by a factor of 1.5, which is common in the existing multi-scale representation methods (*e.g.*, [15, 30]). For compactly, only four scales are used, i.e., 1, 2, 3, and 5. Therefore, the total length of the final PLS feature is 140.

4.1. Configuration

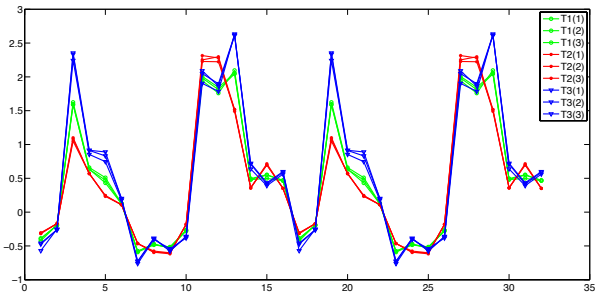
We followed the experimental setup used in [32, 28, 10]. The performance is evaluated in terms of classification accuracy. A fixed-size random subset of images is selected from each class as the training set to train a classifier, and all the remaining images are used as the test set. The classification accuracy is defined as the percentage of the samples correctly classified . The aforementioned process is repeated 100 times and the average classification accuracy is reported. The support vector machine (SVM) implemented by Pontil et al. [26] is used as the classifier with the Gaussian RBF kernel. The cost factor of the SVM is set as the number of images in the dataset, and the shape parameter of the RBF kernel is determined by the standard cross-validation. It is observed that the shape parameter is stable and falls in the range of $[0.25, 0.50]$.

Four challenging texture datasets are selected for the evaluation, including the UMD dataset [4], the UIUC dataset [3], the KTH-TIPS dataset [2], and the ALOT dataset [1]. These datasets have been widely used in the evaluations of many existing texture classification methods. The details of these datasets are summarized in Tab. 1. Our method is compared against several state-of-the-art approaches with reported results on the datasets. Note that not all the methods have available results on each dataset. Thus, the compared methods are not all the same on each dataset. Meanwhile, because the configurations of the datasets are inconsistent, the details of the evaluation on each dataset are slightly different.

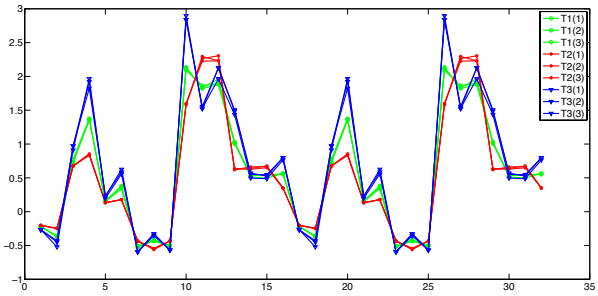
The UMD and UIUC datasets. The experimental configurations on these two datasets are the same. For each class, twenty samples are used for training and the rest for testing. Our method was compared against five methods:

- *(H+L)(S+R)* [12] characterizing texture by the histogram of clustered affine-invariant regions. The affine-invariant region can be seen as one type of image patterns with strong robustness.
- *VG-Fractal* [27], which produces a 13-dimensional lo-

(a) The sample images.



(b) The feature vector collecting all the D(B) values.



(c) The feature vector collecting all the L(B) values.

Figure 4. The PLS features computed on three types of texture. Each type of texture contains three sample images, as shown in (a). Here the corresponding LBP parameters (P, R)s for generating the PLS features are $(4, 1)$, $(8, 1)$, $(4, 2)$, and $(8, 2)$. For the sake of clarity, the values of $D(B)$ and $L(B)$ in Eqn. (7) are collected respectively and shown in (b) and (c). The feature vectors of the texture images of the same type are plotted in the same color.

Table 1. Details of four texture datasets

Dataset	Num. of images	Num. of classes	Resolution	Scale change	Illumination condition
UMD	1000	25	1280×900	significant	uncontrolled
UIUC	1000	25	640×480	significant	uncontrolled
KTH-TIPS	810	10	200×200	small	controlled
ALOT	25000	250	1536×1024	significant	controlled

cal descriptor for each image pattern via local fractal analysis, and builds the texture feature upon the histogram of clusters of the local descriptors.

- *MFS* [29] characterizing the spatial distribution of the image intensity patterns as well as the image gradient patterns by multiple fractal dimensions.

- *OTF* [28] extending the MFS method by using multi-scale oriented templates instead of gradient operator to locate image patterns. A scale alignment strategy using wavelet tight frame decomposition is involved.

- *WMFS* [31, 10], an extension of MFS that replaces the gradient operator with wavelet filters to build up image patterns. A scale normalization scheme based on interest point detection is involved.

Besides, one more method is involved in the UIUC dataset:

- *BIF* [6], defines image patterns based on the partition of the filter-response space of a set of six Gaussian derivative filters.

The KTH-TIPS dataset. The evaluation is the same as above except the compared methods. Besides the aforementioned *(H+L)(S+R)*, *WMFS*, and *BIF* methods for comparison, two histogram-based methods are involved:

- *VZ-MR8* [9], in which the image patterns are characterized by the filter responses of a predefined filter bank.

- *(H+L)(S+S)* [32], an extension of the (H+L)(S+R) method introducing robust local detector and descriptor to represent image patterns.

The ALOT dataset. The evaluation is consistent with that of [10], i.e., the number of the training samples for each class is set as 5, 10,..., 45, 50 respectively. Note that the image resolution of the ALOT dataset is much larger than the other three datasets. To speed up the computation, all the images were down-sampled by half before feature extraction. As there are only a few available results on the ALOT dataset, we focus on the comparisons with three fractal-dimension-based methods. These methods are the aforementioned MFS, OTF, and WMFS methods.

4.2. Results

Table 2 summarizes the performance of the compared methods on the four benchmark datasets. For the number of training samples are not fixed in the evaluation on the ALOT dataset, we only report the classification accuracies generated by using 20 training samples in Tab. 2. It can be seen from Tab. 2 that our approach is very competitive compared with the state-of-the-art methods.

Figures 5, 6, 7 show the per-class classification accuracy achieved by PLS on the UMD, UIUC and KTH-TIPS datasets respectively. As can be seen, more than a half of the classes are 100% correctly classified on the UMD dataset. On the UIUC dataset, the classification accuracy of the class *C14* is the lowest. On the KTH-TIPS dataset, the proposed method performed the worst when dealing with the class *C06*. It is worth noting that the resolution of the images from the KTH-TIPS dataset is much lower than the other datasets. The low resolution might decrease the performance of the fractal-based methods [1], as discussed in [29]. But the proposed method can still achieve excellent performance on the low-resolution dataset due to the robust image pattern representation provided by multi-scale LBPs.

The classification accuracies of the compared methods on the ALOT dataset are shown as curves in Fig. 9. One interesting observation from the figure is that, when the number of the training samples is small, the performance of OTF is better than WMFS, while WMFS outperforms OTF as the number of training samples increases. Note that the scale of the ALOT dataset is much larger than the others. In this case, insufficient training samples might cause instability of classification performance due to the sensitivity of the feature to environmental changes. In contrast, our method performs the best regardless of the number of the training samples. This demonstrates that the proposed PLS feature enjoys both the discriminative ability and the robustness to a wide range of environmental changes. See Fig. 8 for the per-class classification accuracy achieved by PLS on the ALOT dataset when using 50 samples per class for training.

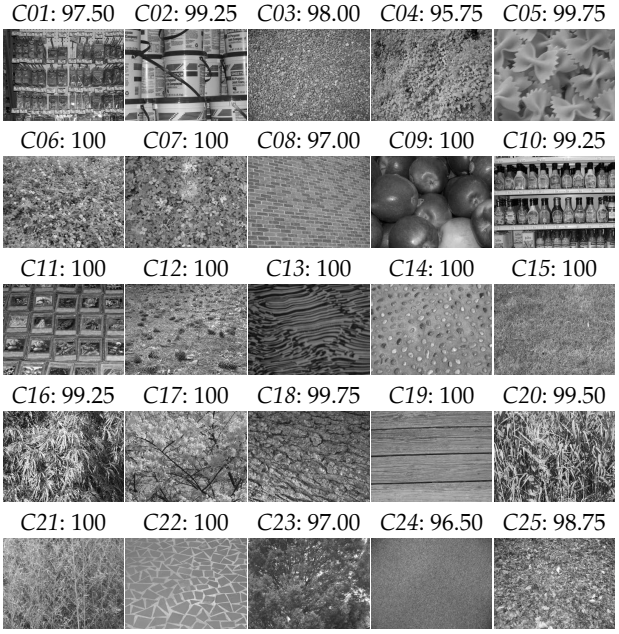


Figure 5. The per-class classification accuracy (%) achieved by PLS on the UMD dataset.

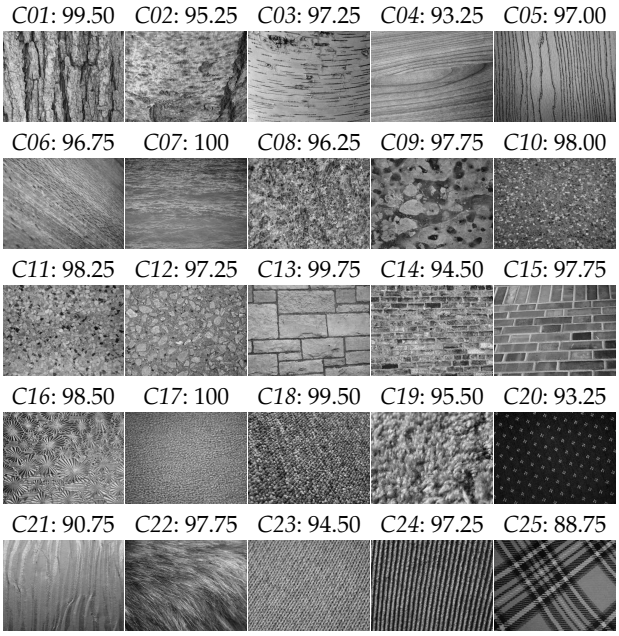


Figure 6. The per-class classification accuracy (%) achieved by PLS on the UIUC dataset.

5. Conclusion

Dense image patterns, such as local binary patterns, provide rich discriminative information for image classification. To integrate the information provided by the image patterns, many existing methods resort to histogram-based statistics. However, such histogram-based methods often lose the spatial details about how the image patterns are distributed. In this paper, based on the concept of lacunarity analysis, a robust texture descriptor called PLS is proposed, which extracts the power-law behaviors of the spatial distribution of local image patterns. The fundamental assumption of our method is that, the local patterns in texture image, if belong to the same type, would exhibit linear behavior with respect to their lacunarity in the log-log coordinates system. Such a linear behavior is characterized by the slope and the

[1] Intuitively, the boxes of the same size used for estimating fractal-related parameters in low-resolution image are less reliable than those in high-resolution image.

Table 2. The classification accuracies (%) on four benchmark datasets.

Dataset	VG-Fractal	MFS	(H+L)(S+R)	VZ-MR8	(H+L)(S+S)	OTF	WMFS	BIF	PLS
UMD	96.36	93.93	96.95	-	-	98.84	98.68	-	**98.99**
UIUC	92.31	92.74	97.02	-	-	98.14	98.60	**98.80**	96.57
KTH-TIPS	-	-	91.30	94.80	96.10	-	96.54	**98.50**	98.40
ALOT	-	78.89	-	-	-	89.33	89.71	-	**93.35**

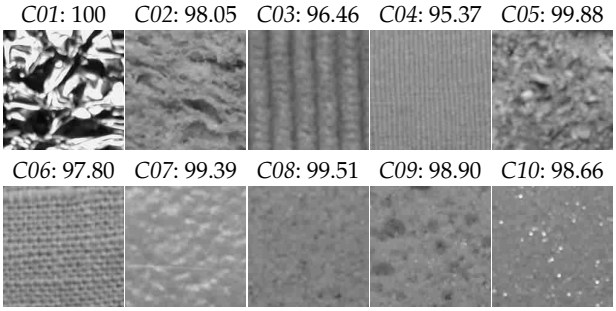


Figure 7. The per-class classification accuracy (%) achieved by PLS on the KTH-TIPS dataset.

intercept in our method. To locate the image patterns as well as to classify them into various types, local binary patterns are employed due to its simplicity and computational efficiency. Experiments on four benchmark datasets have demonstrated the power of our approach. In future, we will analyze the performance of our method using other types of image patterns, and apply our method to scene classification.

Acknowledgement. We would like to thank the area chair and all the reviewers. Y. Sun would like to thank the support by China Scholarship Council Program. Y. Quan and Y. Luo would like to thank the partial support by Singapore MOE Research Grant R-146-000-178-112.

References

[1] ALOT texture dataset. http://www.science.uva.nl/~aloi/public_alot.

[2] KTH-TIPS texture dataset. http://www.nada.kth.se/cvap/databases/kth-tips.

[3] UIUC texture dataset. http://www-cvr.ai.uiuc.edu/ponce_grp/data/index.html.

[4] UMD texture dataset. http://www.cfar.umd.edu/~fer/High-resolution-data-base/hr_database.htm.

[5] C. Allain and M. Cloitre. Characterizing the lacunarity of random and deterministic fractal sets. *Physical review A*, 44(6):3552, 1991.

[6] M. Crosier and L. D. Griffin. Using basic image features for texture classification. *INT J COMPUT VISION*, 88(3):447–460, 2010.

[7] Y. Gefen, Y. Meir, B. B. Mandelbrot, and A. Aharony. Geometric implementation of hypercubic lattices with noninteger dimensionality by use of low lacunarity fractal lattices. *Physical Review Letters*, 50(3):145–148, 1983.

[8] Z. Guo, L. Zhang, and D. Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Trans. Image Process*, 19(6):1657–1663, 2010.

[9] E. Hayman, B. Caputo, M. Fritz, and J.-O. Eklundh. On the significance of real-world conditions for material classification. In *ECCV*, pages 253–266. Springer, 2004.

[10] H. Ji, X. Yang, H. Ling, and Y. Xu. Wavelet domain multi-fractal analysis for static and dynamic texture classification. *IEEE Trans. Image Process*, 2013.

[11] B. Julesz. Textons, the elements of texture perception, and their interactions. *Nature*, 1981.

[12] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Trans. Pattern Anal. Mach. Intell*, 27(8):1265–1278, 2005.

[13] S. Lazebnik, C. Schmid, and J. Ponce. A discriminative framework for texture and object recognition using local image features. In *Toward Category-Level Object Recognition*, pages 423–442. Springer, 2006.

[14] L. Liu, P. Fieguth, G. Kuang, and H. Zha. Sorted random projections for robust texture classification. In *ICCV*, pages 391–398. IEEE, 2011.

[15] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *INT J COMPUT VISION*, 60(2):91–110, 2004.

[16] T. Mäenpää. *The Local Binary Pattern Approach to Texture Analysis: Extenxions and Applications*. Oulun yliopisto, 2003.

[17] Y. Malhi and R. M. Román-Cuesta. Analysis of lacunarity and scales of spatial homogeneity in ikonos images of amazonian tropical forest canopies. *Remote Sensing of Environment*, 112(5):2074–2087, 2008.

[18] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *INT J COMPUT VISION*, 43(1):7–27, 2001.

[19] B. B. Mandelbrot. *The fractal geometry of nature*. Times Books, 1983.

[20] S. W. Myint and N. Lam. A study of lacunarity-based texture analysis approaches to improve urban image classification. *Computers, environment and urban systems*, 29(5):501–523, 2005.

[21] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *PATTERN RECOGN*, 29(1):51–59, 1996.

[22] T. Ojala, M. Pietikäinen, and T. Mäenpää. Gray scale and rotation invariant texture classification with local binary patterns. In *ECCV*, pages 404–420. Springer, 2000.

[23] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. Pattern Anal. Mach. Intell*, 24(7):971–987, 2002.

[24] R. E. Plotnick, R. H. Gardner, W. W. Hargrove, K. Prestegaard, and M. Perlmutter. Lacunarity analysis: a general technique for the analysis of spatial patterns. *Physical review E*, 53(5):5461, 1996.

[25] R. E. Plotnick, R. H. Gardner, and R. V. O'Neill. Lacunarity indices as measures of landscape texture. *LANDSCAPE ECOL*, 8(3):201–211, 1993.

[26] M. Pontil and A. Verri. Support vector machines for 3d object recognition. *IEEE Trans. Pattern Anal. Mach. Intell*, 20(6):637–646, 1998.

Figure 8. The per-class classification accuracy (%) achieved by PLS on the ALOT dataset.

[27] M. Varma and R. Garg. Locally invariant fractal features for statistical texture classification. In *ICCV*, pages 1–8. IEEE, 2007.

[28] Y. Xu, S. Huang, H. Ji, and C. Fermüller. Scale-space texture description on sift-like textons. *COMPUT VIS IMAGE UND*, 2012.

[29] Y. Xu, H. Ji, and C. Fermüller. Viewpoint invariant texture description using fractal analysis. *INT J COMPUT VISION*, 83(1):85–100, 2009.

[30] Y. Xu, Y. Quan, Z. Zhang, H. Ji, C. Fermuller, M. Nishigaki, and D. Dementhon. Contour-based recognition. In *CVPR*, pages 3402–3409. IEEE, 2012.

[31] Y. Xu, X. Yang, H. Ling, and H. Ji. A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid. In *CVPR*, pages 161–168. IEEE, 2010.

[32] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *INT J COMPUT VISION*, 73(2):213–238, 2007.

[33] G. Zhao, T. Ahonen, J. Matas, and M. Pietikainen. Rotation-invariant image and video description with local binary pattern features. *IEEE Trans. Image Process*, 21(4):1465–1477, 2012.

[34] S.-C. Zhu, C.-E. Guo, Y. Wang, and Z. Xu. What are textons? *INT J COMPUT VISION*, 62(1-2):121–143, 2005.

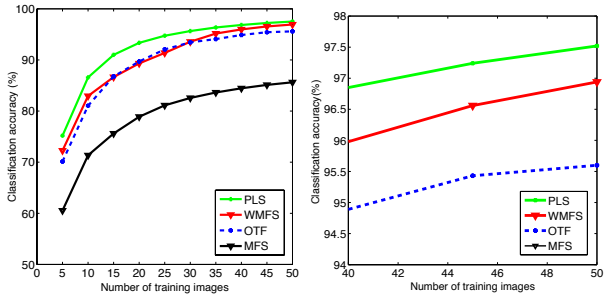


Figure 9. The classification accuracy (%) vs. number of training samples on the ALOT dataset. The classification accuracies are plotted as curves in the left figure, and parts of the curves are resized and shown in the right figure.