Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **5**, September **2018**

BoW-MSN: A Novel Texture Classification Approach

Kun Zhao[1,*], Xue-Cheng Sun[2], Zhe-Ming Lu[3],
Li-Jian Zhou[1], Ting-Yuan Nie[1] and Si-Yuan Hao[1]

[1]School of Information and Control Engineering
Qingdao University of Technology
No.777 Jialingjiang Road, Qingdao, 266520, P.R. China
[*]Corresponding author: sterling1982@163.com

[2]School of Engineering and Applied Science
Columbia University
New York, NY 10027, U.S.A.
xs2278@columbia.edu

[3]School of Aeronautics and Astronautics
Zhejiang University
No.38 Zheda Road, Hangzhou, 310027, P.R. China
zheminglu@zju.edu.cn

Received November, 2017; revised May, 2018

Abstract. *Recently, methods based on complex network theory have shown promise for texture classification tasks. Most of them use global measures of complex networks mapped from texture images to construct feature vectors. This paper presents a novel approach to represent textures as distributions of local measures of multiscale sub-networks of the mapped complex networks. These local measures are called Words of Multiscale Sub-Networks in this paper. Feature vectors are then composed by the words occurrence with the Bag of Words framework. Therefore, the proposed approach is termed as BoW-MSN: Bag of Words of Multiscale Sub-Networks. Experimental results using three real texture data sets of different difficulties have demonstrated the effectiveness of the proposed approach.*

Keywords: Texture Classification, Complex Networks, Multiscale Sub-Networks, Bag of Words

1. **Introduction.** Texture analysis is a basic issue in image processing and computer vision. Although texture can be easily interpreted by humans, designing an automatic tool to perform the same role is still a challenging task. Despite the importance for images, there is no concise definition in literature to the term texture [1]. Fortunately, the absence of a formal definition did not prevent the progress of the research. Many methods of texture analysis have been developed over the years. Most of the existing texture analysis approaches can be roughly divided into following categories: statistical analysis, e.g. co-occurrence matrix [2, 3] and local binary pattern [4, 5]; spectral analysis such us Gabor filters [6, 7] and wavelets [8, 9, 10, 11, 12]; model based analysis, e.g. Gaussian Markov random fields model [13] and fractal geometry model [14, 15]; autonomous entity analysis, e.g. deterministic tourist walks [16, 17, 19]. Recently, some works based on complex networks theory has emerged [18, 19]. However, they extract global measurements from the complex networks, which in some cases, is not effective in texture analysis. Scabini et al. [20] used local information with the Bag of Words framework. However, they

simply use each vertex as a 'visual word', which results in a very high computational cost. Besides, seeing each vertex as independent may miss the local structural information which is important to texture analysis.

The goal of texture analysis is to extract relevant features that present both local and global information of the pixels, ensuring that the whole pattern is correctly represented. Put another way, information of local parts in different scales is necessary to be extracted from textures. In this context, this paper presents a novel approach for texture analysis, namely, Bag of Words of Multiscale Sub-Networks (BoW-MSN), which takes full advantage of the local information in different scales. Instead of the texture images in different resolutions, the 'multiscale' here refers to the different sizes of local parts. The main idea is to build networks from images, and then use the Maximum Excluded Mass Burning (MEMB) algorithm with different 'burning radius' to segment whole networks into sets of sub-networks in different scales so that our BoW-MSN method could be apply over them to generate multiscale features. Instead of global statistical measures from complex networks [18, 19], multiscale measures are extracted to improve the robustness and classification accuracy when both micro and macro textures [18] are present. These multiscale measures are used to build a vocabulary of 'sub-network words' with a BoW framework, and the feature vectors are composed by the sub-network words occurrence in the complex networks. Results on three data sets of different difficulties have shown the effectiveness of the proposed method.

2. **Complex Network and Its Measurements.** Complex network theory, which is frequently applied to the fields of physics and sociology, is an intersection between graph theory and statistics. Owning to its flexibility to model and express different kinds of problems, complex network theory has been gaining in popularity in a wide range. Usually there are two main steps when using complex networks: modelling the problem as networks and extracting measurements from them. To build a complex network with N vertexes, a weighted and undirected graph $\mathsf{G}(\mathsf{V},\mathsf{E})$ is often defined, where $\mathsf{V} = \{v_1, v_2, \dots, v_N\}$ is a set of vertices and $\mathsf{E} = \{\mathbf{e}_{v_i,v_j} | i \neq j\}$ a set of edges with the weight $w(\mathbf{e}_{v_i,v_j})$. In the rest of this article, $\mathbf{e}_{v_i,v_j}$ and $w(\mathbf{e}_{v_i,v_j})$ are remarked as e_{ij} and w_{ij} for brevity. To characterize the structure of complex networks, vertex measures are often used for their powerful ability to describe topological features of networks. For vertex v_i, its degree k_i, weighted degree k_i^w, clustering coefficient c_i and weight distribution difference y_i^w are defined as follows.

$$k_i = \sum_{e_{ij} \in \mathsf{E}} 1 \tag{1}$$

The degree of v_i corresponds to the number of edges attached to v_i.

$$k_i^w = \sum_{e_{ij} \in \mathsf{E}} w_{ij} \tag{2}$$

The weighted degree of v_i is the sum of weights of edges attached to v_i.

$$c_i = \frac{\sum_{e_{im} \in \mathsf{E}, e_{in} \in \mathsf{E}, e_{mn} \in \mathsf{E}} 1}{k_i(k_i - 1)/2} \tag{3}$$

The clustering coefficient of v_i corresponds to a ratio: the number of triangles including v_i to the number of possible triangles centered on v_i. It can be interpreted as the probability for an edge to exist between two randomly picked neighbors of v_i.

$$y_i^w = \sum_{e_{ij} \in \mathsf{E}} (w_{ij}/k_i^w)^2 \tag{4}$$

The weight distribution difference of v_i describes the effect of the difference of edge weight distribution around v_i on the vertex itself.

Based on vertex measures defined above, the corresponding network measures are defined as follows.

$$\overline{k} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} k_i \tag{5}$$

The average degree $\overline{k}$ of network G is the mean of degrees over all n vertexes in G. By parity of reasoning, the average weighted degree, average clustering coefficient and average weight distribution difference of network G can be defined as follows.

$$\overline{k^w} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} k_i^w \tag{6}$$

$$\overline{c} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} c_i \tag{7}$$

$$\overline{y^w} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} y_i^w \tag{8}$$

And the contrast of the measures defined in Equation 1∼4 over all n vertexes in G are calculated as follows.

$$\widehat{k} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} k_i^2 \tag{9}$$

$$\widehat{k^w} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} (k_i^w)^2 \tag{10}$$

$$\widehat{c} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} c_i^2 \tag{11}$$

$$\widehat{y^w} = \frac{1}{n} \sum_{v_i \in \mathsf{G}} (y_j^w)^2 \tag{12}$$

These statistical measures are used to form words of sub-networks at a later step.

3. **Proposed Method.** The proposed method contains three main steps, namely, complex networks generation (3.1), texture signature generation using BoW-MSN (3.2) and texture classification (3.3).

3.1. **Complex Networks Generation for Texture Representation.** Image texture is defined as a bi-dimensional structure of pixels. So in the first step of the proposed method, an image I with $m \times n$ pixels and gray levels between 0 and 255 is modeled as a network $\mathsf{G}(\mathsf{V}, \mathsf{E})$, where $\mathsf{V} = \{v_1, v_2, \ldots, v_{m*n}\}$ is the set of vertices such that each vertex corresponds to one pixel and $\mathsf{E} = \{\mathbf{e}_{v_i,v_j} | i \neq j\}$ is the set of edges. Two vertices v_i and v_j are connected if the Euclidean distance of their corresponding pixels p_i and p_j is smaller than r, namely, the connection radius.

In this stage, as shown in Figure 1 (b), the network presents a regular topology which can not demonstrate the texture variation. Thus, it is necessary to transform this regular network G_r into a series of complex networks $\mathsf{G}_{r,th}$ that have relevant properties for texture analysis, where r is the connection radius and th is a weight threshold. A new complex network is generated from the original regular one when its edges that have weight less than th had been discarded. The edge weight w_{ij} can be calculated by:

$$w_{ij} = \frac{1}{255} |I(p_i) - I(p_j)| \exp(-\frac{d^2_{p_i,p_j}}{2\sigma_r}) \in [0, 1] \tag{13}$$

where d_{p_i,p_j} is the Euclidean distance between pixels that correspond to vertexes v_i and v_j, $I(p_i) \in [0, 255]$ is the intensity of pixel p_i and $\sigma_r \propto r^2$ is the distance factor which affects the changing speed of w_{ij} with d_{p_i,p_j}.

And the strategy for edge discarding can be expressed as:

$$\mathsf{E}_{th} = \mathsf{E} - \bigcup_{\mathbf{e} \in \mathsf{E}} \{\mathbf{e} | w(\mathbf{e}) \leq th\} \tag{14}$$

The threshold affects directly the topology, and can result in networks with dense or sparse connections, as shown in Figure 1 (c). Figure 2 shows two cases of samples from the normalized Brodatz texture data set.

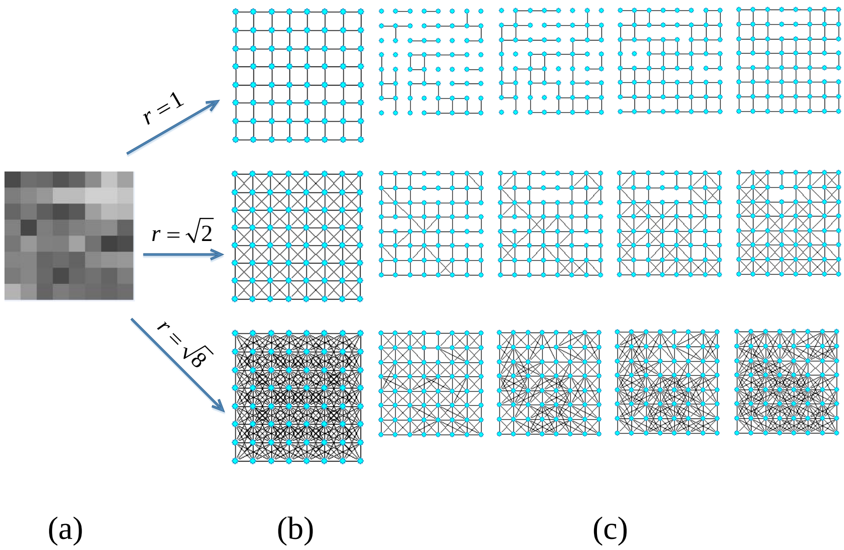

FIGURE 1. Texture image (a) is represented as: (b) regular networks using different r and (c) complex networks with different weight threshold th.

3.2. **Texture Signature Generation using BoW-MSN.** Texture signatures are generated using a Bag of Words of Multiscale Sub-Networks (BoW-MSN) method which came originally from the 'Bag of Words' method applied to document categorization. Its main idea is to represent documents with a histogram of words occurrence along the text.

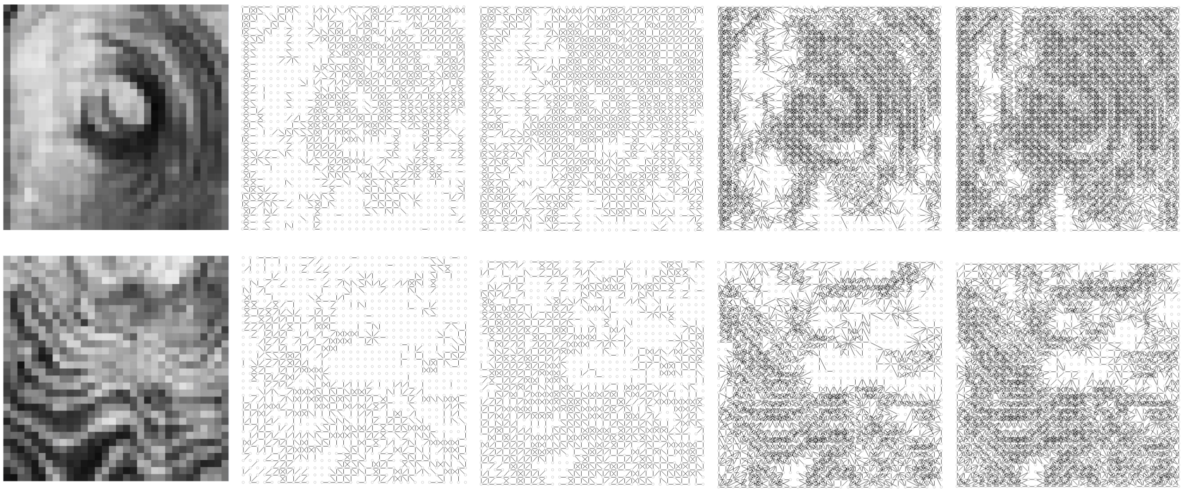

FIGURE 2. Real cases from textures to series of networks: (a) are 32×32 down-sampled images of D43 (top row) and D71 (bottom row) of normalized Brodatz texture data set; (b) and (c) are networks generated with descending thresholds when $r = 2$; (d) and (e) are networks generated with descending thresholds when $r = 3$.

The same idea is used in BoW-MSN. Instead of words literally, local features of complex networks that mapped from texture images are used. Therefore, a sub-network word represents local measures of a region of vertexes with similar properties. Then a vocabulary is build by clustering these sub-network words. At last, the texture signature of an image is expressed as a histogram of words occurrence in the vocabulary. Thus the main steps of BoW-MSN are word construction, vocabulary training and histogram generation.

- Sub-Network Word Construction

The proposed 'sub-network word' should contain local features of the network in different scales. Thus a multiscale network segmentation method is needed, which can not only segment complex network into sub-networks that contain local information, but also reveal the pattern of feature changing in different scales. Fortunately, a complex network has been proved to consist of self repeating patterns on all length scales [21]. To unfold the self-similar properties of a complex network on every length scale, Song et al. segment the whole network using a 'box covering' method with different box size l_B. As shown in Figure 3, all vertices in a box are connected by a minimum distance smaller than the given l_B and thus formed a sub-network. In Figure 3, different sub-networks are marked as different areas with dashed boundary. In this paper, we use the Maximum Excluded Mass Burning (MEMB) [22] algorithm with different burning radius r_B to implement 'box covering' and therefore segment a complex network into sub-networks in different scales. According to [22], $l_B = 2r_B + 1$. So we use r_B as the 'scale factor' of sub-networks.

After a complex network $\mathsf{G}_{r,th}$ was segmented into sub-networks $\mathsf{SG}_{r,th,r_B,1}$, $\mathsf{SG}_{r,th,r_B,2}$, ..., $\mathsf{SG}_{r,th,r_B,i}$, ... with scale r_B, each 'sub-network word' $\overrightarrow{\psi}_{r,th,r_B,i}$ is constructed by a number of measurements defined in section 2. Different configurations with different measurements are marked as the superscripts of $\overrightarrow{\psi}_{r,th,r_B,i}$ as follows.

$$\overrightarrow{\psi}^{1}_{r,th,r_B,i} = \{\overline{k_i}, \overline{c_i}, \overline{y_i^w}, \widehat{k_i}, \widehat{c_i}, \widehat{y_i^w}\} \tag{15}$$

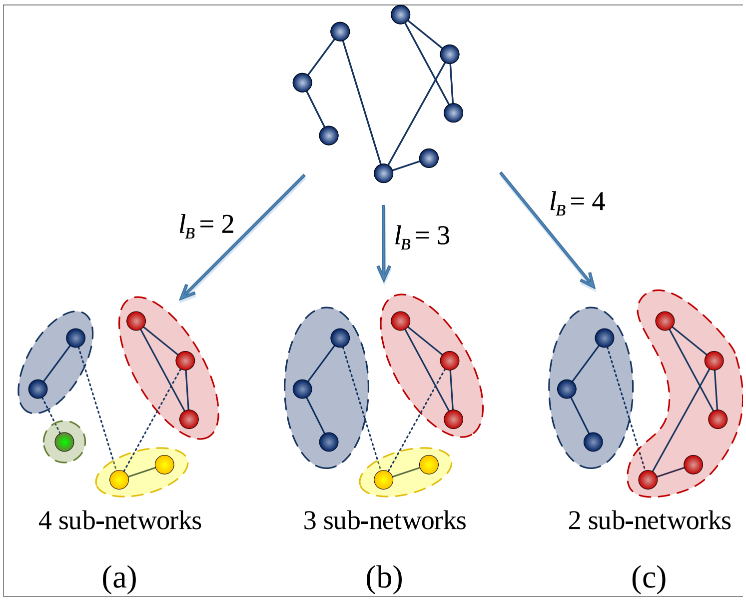

FIGURE 3. Demonstration of the 'box covering' method for different l_B: (a) $l_B = 2$, (b) $l_B = 3$ and (c) $l_B = 4$.

$$\overrightarrow{\psi}^{2}_{r,th,r_B,i} = \{\overline{k_i}, \overline{c_i}, \overline{y_i^w}, \widehat{k_i}, \widehat{k_i^w}, \widehat{y_i^w}\} \tag{16}$$

$$\overrightarrow{\psi}^{3}_{r,th,r_B,i} = \{\overline{k_i}, \overline{k_i^w}, \overline{y_i^w}, \widehat{k_i}, \widehat{k_i^w}, \widehat{y_i^w}\} \tag{17}$$

- Vocabulary Training

All the sub-network words $\overrightarrow{\psi}_{r,th,r_B,i}$ with different r, th, r_B of all texture images in the training set converge into a bag of words. Then a k-means clustering algorithm is performed in the bag to train a vocabulary. The algorithm is initialized randomly and performs a clustering based on the Euclidean distance of the sub-network words vectors, returning $\mathsf{C} = \{C_1, ..., C_k\}$ centers corresponding to the sub-network words in the vocabulary. With the centers, it is possible to assign an index $i \in [1, k]$ to each sub-network word that is not in the vocabulary by finding the nearest centers C_i. Thus, every sub-network is labeled by an index.

- Histogram Generation

Any texture image can be expressed as a set of sub-networks with fixed indexes. We then obtain the feature vector of the texture image by

$$\overrightarrow{\chi} = \{\widetilde{h_1}, \widetilde{h_2}, ..., \widetilde{h_k}\} \tag{18}$$

where $\widetilde{h_i} = h_i / \max_{j\in[1,k]}(h_j)$ is the normalized histogram of the index i.

3.3. **Texture Classification.** Our feature vector analysis is carried out by applying a Linear Discriminant Analysis (LDA) to the data. In the classification, the Nearest Neighbours classifier (NNC) is used. We have chosen a simple classifier rather than a sophisticated one in order to highlight the importance of texture descriptors.

4. Experimental Results and Discussion.

4.1. **Experimental Data Sets.** In order to evaluate the proposed method, the signatures were calculated for appropriate configurations for texture classification. Three texture data sets are used in this evaluation.

Outex TC10 [23]**.** This data set is used to evaluate the rotation invariance of algorithms. It contains 24 classes of texture images captured under 9 rotation angles (0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, and 90°) and three different illuminations (horizon, inca, and t184). There are 20 images with the resolution of 128×128 for each rotation angle. The 20 images of illuminations "inca" with the resolution of 100 pdi and rotation angle 0° in each class were adopted as the training data and the 160 images of other 8 rotation angles are used for test. Figure 4 shows some sample images from Outex TC10 data set.

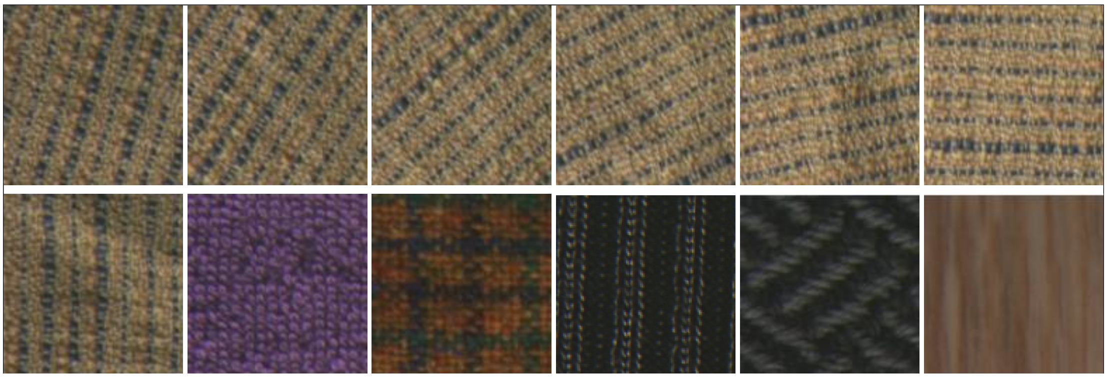

Figure 4. Samples from Outex TC10: the top row shows the same sample with different rotation angles, and the bottom row shows samples from different categories (0°).

CUReT [24]**.** This data set is used to evaluate the illumination invariance of algorithms. It contains 61 texture classes each with 92 selected images (200×200) under different illumination direction. We randomly choose N texture images from each class as training samples, while the remaining $92 - N$ images for test. Figure 5 shows some sample images from CUReT data set.

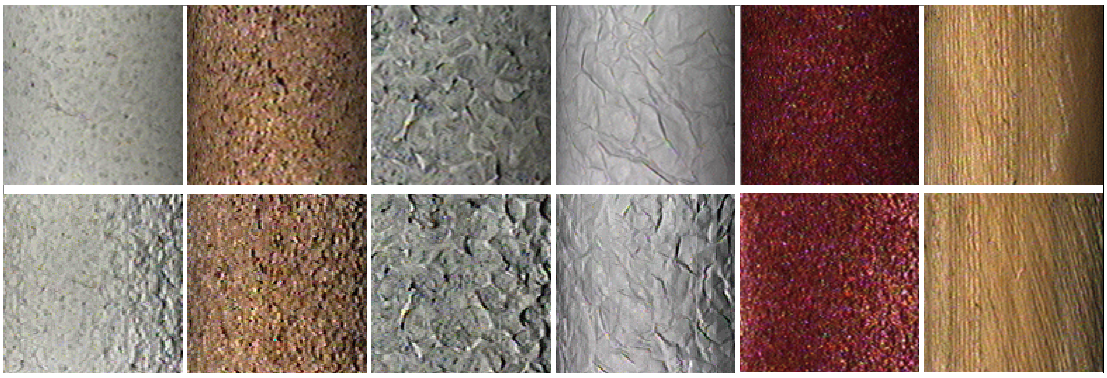

Figure 5. Samples from CUReT: the top row shows samples from different categories, and the bottom row shows the same samples under another illumination.

UIUC [25]**.** This data set is used to evaluate performance of algorithms under the condition of complex variations. It contains 25 classes with 40 images (640×480) each, captured under varying viewpoints. We randomly choose N texture images from each class as training samples, while the remaining $40 - N$ images for test. Figure 6 shows some sample images from UIUC data set.

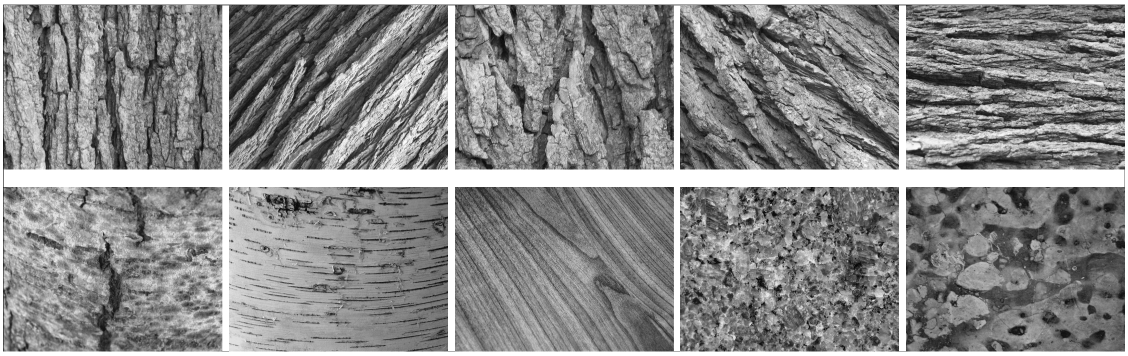

FIGURE 6. Samples from UIUC: the top row shows samples from the same category, and the bottom row shows the samples from different categories.

4.2. **Parameter Evaluation.** From section 3 we can learn that there are totally 5 key parameters of the proposed method to be configured: the connection radius r, the distance factor σ_r and the weight threshold th in the step of complex networks generation; the burning radius r_B in the step of sub-networks construction; and the index number k in the step of vocabulary training. Therefore, it is necessary to evaluate the behavior of the method concerning different configurations of these parameters.

- Parameters Configuration at the Step of Complex Networks Generation

When generating networks from texture images, an appropriate connection radius r should firstly be configured. We set r depend on the size of the target image. Therefor, we set $r = 2$ for Outex TC10 (128×128), $r = \sqrt{8}$ for CUReT (200×200) and $r = 3$ for UIUC (640×480). From Equation 13 we can learn that $\sigma_r \propto r^2$. In this paper, we set $\sigma_r = \frac{1}{2}r^2$ in all situations. For the configuration of weight thresholds th, inspired by literature [27], we use a similar strategy to select thresholds automatically. The different is that we use the contrast of the degree and weighted degree distributions (two average curves) instead of the energy of the degree distributions (one average curve) in [27] because the the measures that used to construct the final sub-network words of the proposed method are mainly based on the contrast of degree and weighted degree of sub-networks.

- Parameters Configuration at the Step of Sub-Networks Construction

The burning radius r_B is the scale factor when a mapped complex network being segmented into a number of sub-networks. We set this parameter empirically by evaluating the final success rate of classification. Table 1 summarizes the results for 6 different sets of scales (S1$\sim$S6). To compose each set of multiscale burning radius r_B, we considered an initial scale (r_{B1}), which is gradually increased by a scale Δr_B, until it achieves a maximum scale (r_{Bmax}). For each set, we put local features in Equation 17 at all scales together into a "bag" to train the vocabulary.

- Parameters Configuration at the Step of Vocabulary Training

We set k cluster centers when using the k-means algorithm during the stage of vocabulary training. The clustering stops when it iterated 100 times or the error was less than 0.009. Thus, k descriptors in total to express a texture image. We set k empirically also by evaluating the final success rate of classification. Table 2 summarizes the results for 5 different numbers of cluster centers.

4.3. **Comparison With Other Methods.** After evaluating different configurations, we choose the parameters which lead to the best performance on the three data sets, namely, Outex TC10, CUReT and UIUC. The proposed approach is compared with other well-known descriptors: LBP [4], Circular Gabor [7], LTP [26], CLBP [5], Multi-scale GLCM

Table 1. Results on three data sets for the proposed method for different configurations of scales.

Set	Scales			Success rate (%)		
	r_{B1}	Δr_B	r_{Bmax}	Outex TC10	CUReT (46:46)	UIUC (20:20)
S1	2	1	4	**99.27**	92.29	81.16
S2	3	1	5	94.78	**96.16**	78.38
S3	4	1	6	83.02	90.08	74.40
S4	2	2	6	96.33	89.72	**90.93**
S5	3	2	7	89.45	91.29	86.42
S6	4	2	8	76.17	86.48	80.84

Table 2. Results on three data sets for the proposed method for different cluster centers.

k	Success rate (%)		
	Outex TC10	CUReT (46:46)	UIUC (20:20)
50	98.31	94.87	86.33
100	**99.27**	**96.16**	90.77
200	99.14	94.47	**90.93**
500	98.86	93.98	88.56
1000	98.02	95.17	88.94

[3] and existing complex network based methods including Complex Networks (CN) [18], Deterministic Walks on Complex Networks (DS+CN) [19] and Complex Network Descriptors using Bag of Words Framework (BoW+CN) [20]. The results are shown in Table 3∼5. The superscripts of the proposed BoW-MSN approach denote the corresponding configurations in Equation 15∼17.

- Performance Evaluation

In the first experiment, Outex TC10 data set is used to evaluate the rotation invariance of algorithms. The results are shown in Table 3.

Table 3. Texture classification results (%) achieved by proposed and other well-known descriptors on Outex TC10 data set. Top 3 results of each data set are in bold and the best one is in underline & bold.

Method	Success rate (%)	Method	Success rate (%)
LBP	98.15	CN	89.67
C-Gabor	81.42	DW+CN	90.08
LTP	**98.64**	BoW+CN	85.47
M-GLCM	89.45	BoW-MSN^2	98.35
CLBP	<u>**99.45**</u>	BoW-MSN^3	**99.27**

In the second experiment, CUReT data set is used to evaluate the illumination invariance of algorithms. The results under the conditions of different numbers of training and test samples are shown in Table 4.

Table 4. Texture classification results (%) achieved by proposed and other well-known descriptors on CUReT data set. Top 3 results of each data set are in bold and the best one is in underline & bold.

Method	Success rates (%)					
	46:46	41:51	37:55	32:60	28:64	23:69
LBP	**94.20**	**93.82**	**93.03**	91.51	88.44	83.63
C-Gabor	79.09	78.54	77.76	76.21	74.05	70.33
LTP	92.66	92.03	91.21	89.92	87.81	85.15
M-GLCM	86.91	86.38	85.65	84.01	80.33	74.20
CLBP	<u>**97.33**</u>	<u>**97.09**</u>	<u>**96.43**</u>	<u>**95.42**</u>	**93.85**	**91.77**
CN	87.39	87.01	86.49	85.22	82.03	76.46
DW+CN	91.23	90.76	90.00	88.86	86.67	82.91
BoW+CN	82.06	81.71	81.17	80.43	79.31	77.67
BoW-MSN^1	91.01	90.83	90.52	89.93	89.01	87.67
BoW-MSN^2	93.37	93.15	92.80	**92.28**	**91.50**	**90.25**
BoW-MSN^3	**96.16**	**95.98**	**95.72**	**95.23**	<u>**94.41**</u>	<u>**93.05**</u>

In the third experiment, UIUC data set is used to evaluate performance of algorithms under the condition of complex variations. The results of different numbers of training and test samples are shown in Table 5.

Table 5. Texture classification results (%) achieved by proposed and other well-known descriptors on UIUC data set. Top 3 results of each data set are in bold and the best one is in underline & bold.

Method	Success rates (%)					
	20:20	19:21	18:22	17:23	16:24	15:25
LBP	79.01	78.47	77.55	76.18	74.31	71.94
C-Gabor	72.02	71.76	70.92	69.58	67.33	63.74
LTP	82.34	82.08	81.67	81.11	80.21	79.10
M-GLCM	81.72	81.01	79.55	77.25	73.67	68.81
CLBP	<u>**91.19**</u>	<u>**90.83**</u>	<u>**90.29**</u>	**89.60**	**88.67**	**87.42**
CN	78.73	78.33	77.42	76.01	73.77	70.21
DW+CN	80.10	79.73	79.00	77.57	75.09	71.03
BoW+CN	76.29	75.99	75.45	74.65	73.37	71.55
BoW-MSN^1	83.16	82.98	82.64	82.15	81.37	80.33
BoW-MSN^2	**87.50**	**87.23**	**86.79**	**86.19**	**85.26**	**84.02**
BoW-MSN^3	**90.93**	**90.67**	**90.23**	<u>**89.65**</u>	<u>**88.78**</u>	<u>**87.58**</u>

- Discussion

From the distribution of data in bold in Table 3∼5 we can learn that both CLBP and proposed BoW-MSN descriptors achieve good performance in all tests. LTP and LBP do well with sufficient training data but present a significant drop when training samples were reduced. On the contrary, the proposed BoW-MSN descriptors performs well with a few training samples thanks to the multiscale local information that it used. A texture image was segmented into parts in several ways by different scales (as shown in Figure 3), which can be considered as a method to add training samples.

When compared with other traditional vertex degree based complex network descriptors (CN, DW+CN and BoW+CN), the proposed BoW-MSN descriptors present a excellent texture discrimination thanks to the new networks generation method (Equation 13, Equation 14) and the usage of more comprehensive measures defined in Equation 7, Equation 8, Equation 11 and Equation 12. In the case of using a same Bag of Words framework, BoW-MSN descriptors win out overcoming BoW+CN descriptors because the multiscale sub-networks we used keep the information of local structure which is very important to texture analysis.

5. **Conclusion.** In this work, a novel method for texture classification was proposed. A multiscale segmentation method for complex networks has been used to form the local features of textures. A BoW-MSN approach has then been used to build the final global feature for each texture image. Results show that the proposed method overcomes traditional methods in challenging databases and also maintains a competitive advantage in the situation of a few training samples for widely used cases.

As future works, we intend to use more effective statistical frameworks such as Soft Coding Feature model to fuse local texture information. Besides, a multiresolution analysis should be explored in the context of complex network based approaches.

Acknowledgment. Part of this work was supported by following grant respectively: Qingdao Source Innovation Program (Special Project for Applied Research - Special Project for Young Scholars): No. 17-1-1-2-jch; Natural Science Foundation of Shandong Province, China: No. ZR2017PF004, No. ZR2016FQ14 and No. ZR2016FP06; National Natural Science Foundation of China: No. 61572269, No. 61501278, No. 61701272 and No. 61771271.

REFERENCES

[1] J. Zhang and T. Tan, Brief review of invariant texture analysis methods, *Pattern Recognition*, vol.35, no.3, pp.735-747, 2002.

[2] R. M. Haralick, K. Shanmugam and I. Dinstein, Textural features for image classification, *IEEE Transactions on Systems, Man and Cybernetics*, vol.3, no.6, pp.610-621, 1973.

[3] F. R. D. Siqueira, W. R. Schwartz and H. Pedrini, Multi-scale gray level co-occurrence matrices for texture description, *Neurocomputing*, vol.120, no.10, pp.336-345, 2013.

[4] T. Ojala, M. Pietikäinen and T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.24, no.7, pp.971-987, 2002.

[5] Z. Guo, L. Zhang and D. Zhang, A completed modeling of local binary pattern operator for texture classification, *IEEE Transactions on Image Processing*, vol.19, no.6, pp.1657-1663, 2010.

[6] A. K. Jain and F. Farrokhnia, Unsupervised texture segmentation using gabor filters, *Pattern Recognition*, vol.24, no.12, pp.1167-1186, 1991.

[7] G. Haley, B. Manjunath, Rotation-invariant texture classification using a complete space-frequency model, *IEEE Transactions on Image Processing*, vol.8, no.2, pp.255-269, 1999.

[8] E. P. Simoncelli and J. Portilla, Texture characterization via joint statistics of wavelet coefficient magnitudes, *Proc. of 1998 IEEE International Conference on Image Processing*, Chicago, U.S.A., vol.1, pp.62-66, 1998.

[9] C. M. Pun and M. C. Lee, Log-polar wavelet energy signatures for rotation and scale invariant texture classification, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.25, no.5, pp.590-603, 2003.

[10] Y. L. Qiao, Z. M. Lu, J. S. Pan and S. H. Sun, Fast k-nearest neighbor search algorithm based on pyramid structure of wavelet transform and its application to texture classification, *Digital Signal Processing: A Review Journal*, vol.20, no.3, pp.837-845, 2010.

[11] Y. L. Qiao, Z. M. Lu, J. S. Pan and S. H. Sun, Spline wavelets based texture features for image retrieval, *International Journal of Innovative Computing, Information and Control*, vol.2, no.3, pp.653-658, 2006.
[12] J. W. Wang, C. H. Chen and J. S. Pan, Genetic feature selection for texture classification using 2-D non-separable wavelet bases, *IEICE Transactions on Fundamentals of Electronics, Communications And Computer Sciences*, vol.E81-A, no.8, pp.1635-1655, 1998.
[13] F. S. Cohen, Z. Fan and M. A. Patel, Classification of rotated and scaled textured images using Gaussian Markov random field models, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.13, no.2, pp.192-202, 1991.
[14] Y. Xu, H. Ji and C. Fermüler, Viewpoint invariant texture description using fractal analysis, *International Journal of Computer Vision*, vol.83, no.1, pp.85-100, 2009.
[15] M. Varma and R. Garg, Locally invariant fractal features for statistical texture classification, *Proc. of 2007 IEEE International Conference on Computer Vision*, Rio de Janeiro, Brazil, pp.1-8, 2007.
[16] A. R. Backes, W. N. Gonçalves, A. S. Martinez and O. M. Bruno, Texture analysis and classification using deterministic tourist walk, *Pattern Recognition*, vol.43, no.3, pp.685-694, 2010.
[17] W. N. Gonçalves and O. M. Bruno, Combining fractal and deterministic walkers for texture analysis and classification, *Pattern Recognition*, vol.46, no.11, pp.2953-2968, 2013.
[18] A. R. Backes, D. Casanova and O. M. Bruno, Texture analysis and classification: A complex network-based approach, *Information Sciences*, vol.219, no.8, pp.168-180, 2013.
[19] L. N. Couto, T. P. Ribeiro and A. R. Backes, Texture characterization via improved deterministic walks on image-generated complex network, *Proc. of 2015 IEEE International Conference on Image Processing*, Québec City, Canada, pp.4416-4420, 2015.
[20] L. F. S. Scabini, W. N. Gonçalves and A. A. Castro Jr., Texture analysis by bag-of-visual-words of complex networks, *Proc. of 20th Iberoamerican Congress on Pattern Recognition*, Montevideo, Uruguay, pp.485-492, 2015.
[21] C. Song, S. Havlin and H. A. Makse, Self-similarity of complex networks, *Nature*, vol.433, no.7024, pp.392-395, 2005.
[22] C. Song, L. K. Gallos and S. Havlin, How to calculate the fractal dimension of a complex network: the box covering algorithm, *Journal of Statistical Mechanics: Theory and Experiment*, vol.2007, no.3, pp.297-316, 2007.
[23] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen and S. Huovinen, Outex - new framework for empirical evaluation of texture analysis algorithms, *Proc. of the 16th International Conference on Pattern Recognition*, Québec City, Canada, pp.701-706, 2002.
[24] K. J. Dana, B. Van Ginneken, S. K. Nayar and J. J. Koenderink, Reflectance and texture of real-Cworld surfaces, *ACM Transactions on Graphics*, vol.18, no.1, pp.1-34, 1999.
[25] S. Lazebnik, C. Schmid and J. Ponce, A sparse texture representation using local affine regions, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol.27, no.8, pp.1265-1278, 2005.
[26] X. Tan and B. Triggs, Enhanced local texture feature sets for face recognition under difficult lighting conditions, *IEEE Transactions on Image Processing*, vol.19, no.6, pp.1635-1650, 2010.
[27] T. P. Ribeiro, L. N. Couto, A. R. Backes and C. A. Z. Barcelos, Texture characterization via automatic threshold selection on image-generated complex network, *Progress in Pattern Recognition, Image Analysis, Computer Vision and Applications*, Springer International Publishing, pp.468-476, 2015.