Journal of Information Hiding and Multimedia Signal Processing ©2015 ISSN 2073-4212
Ubiquitous International Volume **6**, Number **3**, May **2015**

Rotation Invariant Texture Classification Using Principal Direction Estimation and Random Projection

Yulong Qiao and Ying Zhao*

School of Information and Communication Engineering
Harbin Engineering University
Harbin, 150001, China
*corresponding author: wszy198979wszy@163.com

Received November, 2014; revised December, 2014

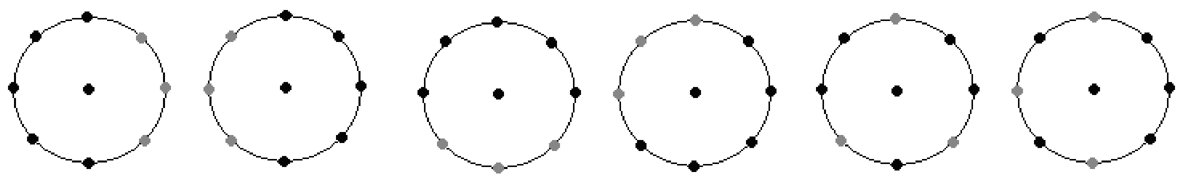
Abstract. *The rotation invariant texture classification is an important application of texture analysis. A rotated texture is often perceived by the changed dominant direction. This paper proposes an effective rotation-invariant texture classification method by combining the local patch based method with the orientation estimation. For a texture sample, the Principal component analysis is applied to its local patch to estimate the local orientation, and then the dominant orientation is determined with the maximum value of the local orientation distribution. In order to extract the feature vector, each local patch is rotated along the dominant orientation after circular interpolation. By using the random projection, the local gray value vector of a patch is mapped into a low-dimensional feature vector that is placed in the bag of words model, together with local orientation feature. The simulation experiments are performed on two benchmark texture databases Outex and CUReT datasets. The results demonstrate the proposed method has a comparable performance with the existing methods.*
Keywords: Rotation-invariant texture classification, Principal component analysis, Orientation estimation

1. **Introduction.** Texture is a nature attribute of the images from the real world. Texture classification is to classify different texture categories due to our priori knowledge. It's a common issue in computer vision and image processing, and has been widely used in many fields. In the texture classification field, classify textures with rotation invariant is an essential need in many applications.
Conventional method includes polar coordinate transform, Fourier transform, High-order statistics, center-symmetric auto-correlation, local binary pattern[1, 2], Gaussian. Markov random fields, Radon transform[3, 4], Wavelet transform[5], and so forth[6]. In recent years, many researches have been focused on the rotation invariant texture classification (RITC) issue. It is demonstrated that the joint distribution of intensity values over extremely compact neighborhoods outperforms classification using filter banks with large support by Varma in [7]. VZ_joint[8] is a patch based method proposed by Varma. To achieve rotation invariant property, the author points out to find a local direction and arrange the intensity values from each circle in local patch. Although patch based method can achieve high accuracy classification rate, the computational complexity is also very high. The problem is solved in [9], which uses the random projection to map the high-dimensional feature into low dimensional with random measure matrix. In [10], Zhen uses

a structure tensor to estimate the local orientation as a comparison method. The author also proposes a new method—joint_sort, where the intensity values from local patch are sorted in a descending order. It has a lower accuracy but save the time using to find local direction. However, both of them ignore the global direction. As shown in Fig.1, as the condition patches with size of 3 pixels, in the patches feature extraction stage, there will be no difference between Fig.1(a) an Fig.1(b) in the method of VZ_joint. The same situation also happen between Fig.1(a) and Fig.1(c) in the method of joint_sort. Although the overlapping patches can support some orientation information, they don't have enough information of textures with strong global direction.

Textures are rich in orientation information, which can be divided into local orientation and global orientation. As shown in Fig.2, the white rings point out the local orientation and the black arrows show the mainly global orientation. The local orientation region shows the microscopic structure of texture image, and the global orientation provides the global structure. In order to keep the global and local orientation information at the same time, this paper presents a method combining the local patch and orientation estimate together. The Principal component analysis theory is used to find the local orientation of the micro patch, and the mainly orientation of the texture image is found by the local orientation distribution. Then the feature vectors extracted from each patch are arranged in the mainly orientation, after random projection, the low feature vector and local orientation feature of each patch are used for clustering, training and testing. Experiments have been taken on Outex and CUReT database, which prove the effectiveness of the proposed method.

The rest of this paper is organized as follow: In section 2, we give a brief review of joint_sort and VZ_joint method, and present the method used to estimate the orientation in textures. In section 3, we introduce the proposed method. In section 4, we present the experimental results on benchmark texture datasets. Finally, we draw the conclusions in section 5.



(a) (b) (c)

FIGURE 1. Three images with size of 3×6 pixels

(a) (b)

FIGURE 2. Two texture images

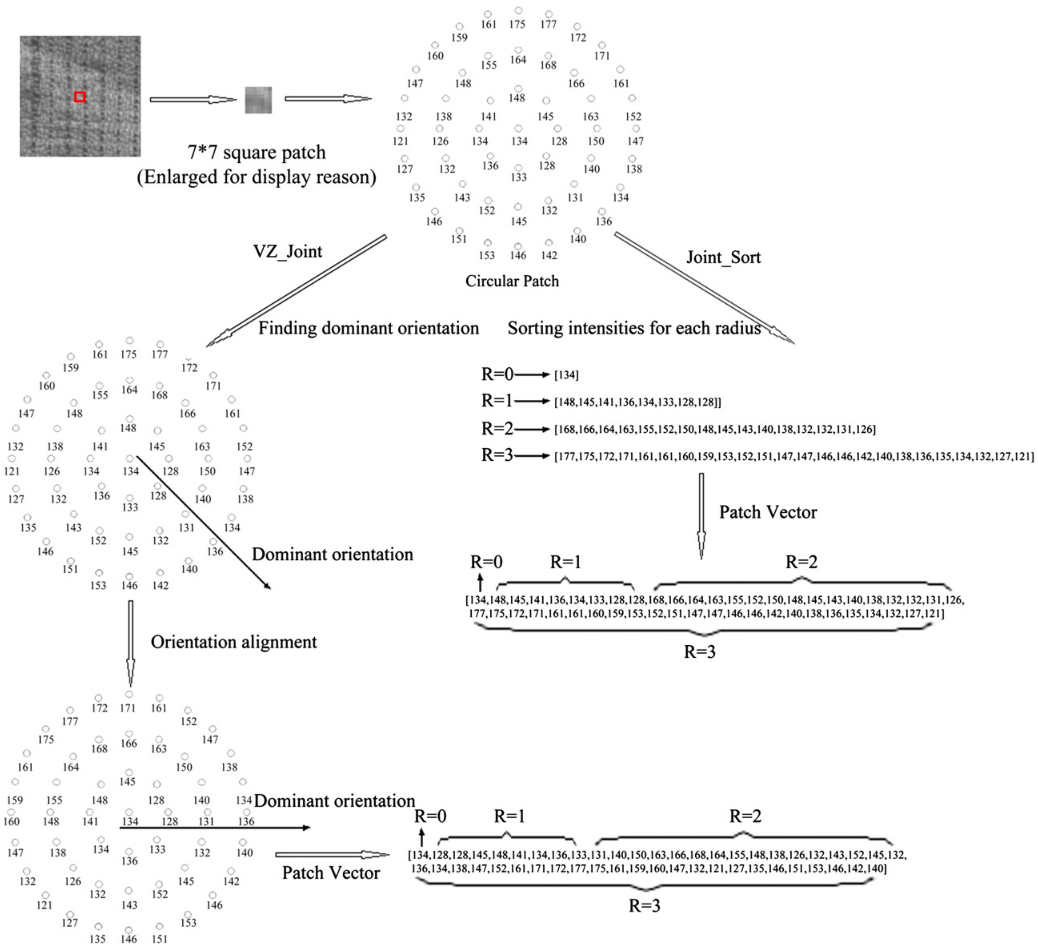


FIGURE 3. The difference in feature extraction between VZ_joint and Joint_sort (Source: [10])

2. Backgrounds.

2.1. Brief review of VZ_joint and Joint_sort. VZ_joint is a patch-based method proposed by Varma. The method takes the $N \times N$ square local overlapping patches from texture image to get the N^2 dimensional feature vectors. Then a textons library is built from the train set by clustering and textons histograms are used to classify textures. To address the rotation invariant property, a circle neighborhood is used instead of square in [11], and a local direction achieved by structure tensor[10] is found, then feature vectors are arranged in this direction.
Similar framework is to be found in joint_sort method. The only difference between them is in the feature extraction stage. In joint_sort, the intensity values in each concentric circle are sorted in a descending order, and then concatenate together as a long vector to present the patch. Fig.3 shows the detail of difference between the two methods.

2.2. The Methods of Orientation Estimate. In [12], an orientation estimation method based on Principal component analysis (PCA) and multiscale pyramid decomposition is proposed. The PCA analysis is applied to find the Maximum Likelihood (ML) estimate of the local orientation[12]. For an image, its gradient image at (x_k, y_k) is denoted as$\nabla f(k) = [\partial f(x_k, y_k)/\partial x, \partial f(x_k, y_k)/\partial y]$, then the gradient image is divided into local blocks (overlapped or non-overlapped). For each $N \times N$ block, we arrange the gradient values by rows to get a $N^2 \times 2$ matrix $\mathbf{G}$, which can be denoted as follows:

$$\mathbf{G} = \begin{bmatrix} \nabla f(1) \\ \vdots \\ \nabla f(N) \end{bmatrix} \tag{1}$$

Then the Singular Value Decomposition (SVD) is applied on matrix $\mathbf{G}$:

$$\mathbf{G} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{2}$$

And the first column v_1 of $\mathbf{V}$ can be seen as the dominant orientation of the gradient field. Besides, a measure R of accuracy of dominance of the estimate is computed as follows:

$$R = \frac{s_1 - s_2}{s_1 + s_2} \tag{3}$$

More detail information of the algorithm can be found in [13]. And before the orientation estimation, the Gaussian low-pass filter is used to avoid the effects of noise. In this paper, to simplify the algorithm, instead of combining the PCA with the multiscale pyramid decomposition, we only use the PCA to estimate the local orientation. From the distribution of local orientation we can get the mainly orientation as the global orientation. And the accuracy measure R is also used as a local feature.

3. **The Proposed Patch_orientation Method.**

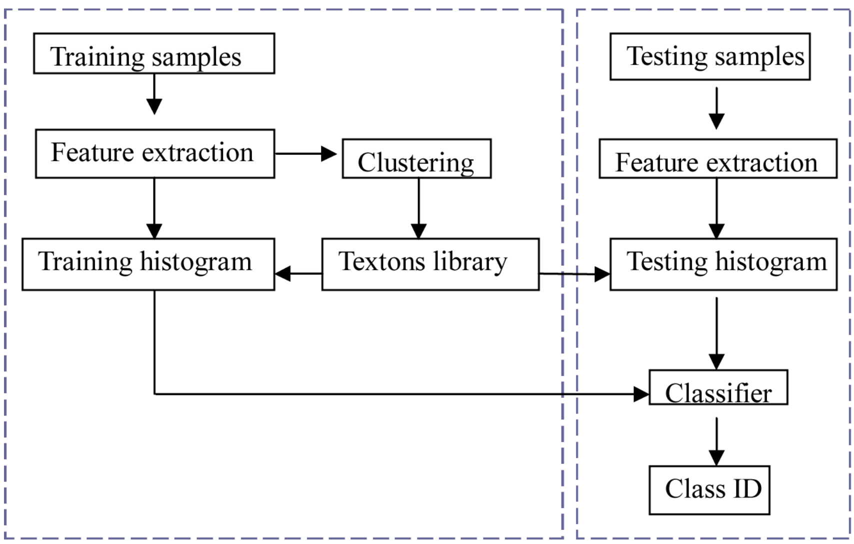


FIGURE 4. Patch based texture classification method framework

3.1. **Patch Based Texture Classification Method Framework.** The patch based texture classification method can be mainly divided into two parts: feature extraction and classification. For the training samples, the overlapping patches are extracted, which are used to estimate the local orientation and form the local feature with the mainly orientation of the whole image. After random projections, the textons library is built from low-dimensional feature vectors by k-means clustering. Then the statistical textons distribution histogram of each texture class can represent this class. For example, if there

are C texture categories with D samples per class, and each class has K textons, then the textons number of the texton dictionary is $C \times K$. And the D training histograms with the dimension of $C \times K$ will be seen as the representation of this texture class.
The classifier employed in this paper is also the K-nearest neighbor (KNN). In this theory, according to the similarity measure, the category label of the testing sample is decided by the majority of k nearest neighbors in training samples belong. In this paper, the k is set to be one, and we use the χ^2 statistics as the similarity measure. Assume $\mathbf{H}_t$ and $\mathbf{H}_r$ to represent two N-dimensional histograms, then the distance D between them can be computed like this:

$$D_{\chi^2}(H_t, H_r) = \sum_{i=1}^{N} \frac{[\mathbf{H}_t(i) - \mathbf{H}_r(i)]^2}{[\mathbf{H}_t(i) - \mathbf{H}_r(i)]} \tag{4}$$

3.2. **Feature Extractions.** In this stage, the grey value feature and local orientation feature are extracted in the same time. As shown in Fig.5, for $patch_1$, after orientation estimation, the local angle α_1 and estimation measure R_1 are computed from $patch_1$. Then all local angles from the image generate the orientation histogram, from which we can get the total angle β on the peak of the histogram. The points in a local patch are interpolated along the circle rings.

$$\mathbf{x} = [x_{i,j}], 1 \leq i \leq r, j \in [1, 8i] \tag{5}$$

Here r is the total number of the ring, and $x_{i,j}$ is the jth point on the ith ring. The first point of each ring is beginning with the β direction. Then we can get the high dimensional vector $\mathbf{x}$ by arranging the value circle by circle. To reduce the computational complexity, the random projection is applied to mapping the high dimension patch vector into a low dimension one. The random projection theory is straightforward: For a vector $\mathbf{x} = [x_1,\ x_2,,\ x_N]^T$, choose a random matrix $\Phi = [\varphi_{i,j}]_{M\times N}$, the random projection is

$$\mathbf{y} = \Phi\mathbf{x} \tag{6}$$

Where $N \gg M$, and $\mathbf{y} = [y_1,\ y_2,,\ y_M]^T$ is the low-dimensional vector. In [9], inspired by the theory of compressed sensing, the author presents a texture classification method using random projection, where Φ is an independent, zero-mean and unit-variance Gaussian random matrix. As mentioned in [14], there are three kinds of random matrices in compress sensing: Gaussian measurements, Binary measurements, Fourier measurements. According to the experiment results, we choose the matrix Φ to be an independent random matrix with the symmetric Bernoulli distribution $P(\Phi_{Mi} = \pm 1/\sqrt{M}) = 1/2$.
In order to stress the orientation information, after random projection, we combine the low dimensional vector with the local angle α_1 and estimation measure R_1, which are used as the final feature vector. (The local angle α_1 is normalized between 0 and 1.) Then the texture feature vector can be trained and tested under the framework in section 3.1. The experiments under large texture database demonstrate the patch_orientation method (OP method) has a better performance than joint_sort method.

4. **Experimental Results.** To evaluate the effectiveness of the proposed method, we carried out a series of experiments compared with joint_sort and VZ_joint on two benchmark texture datasets: Outex database and Columbia-Utrecht Reflection and Texture

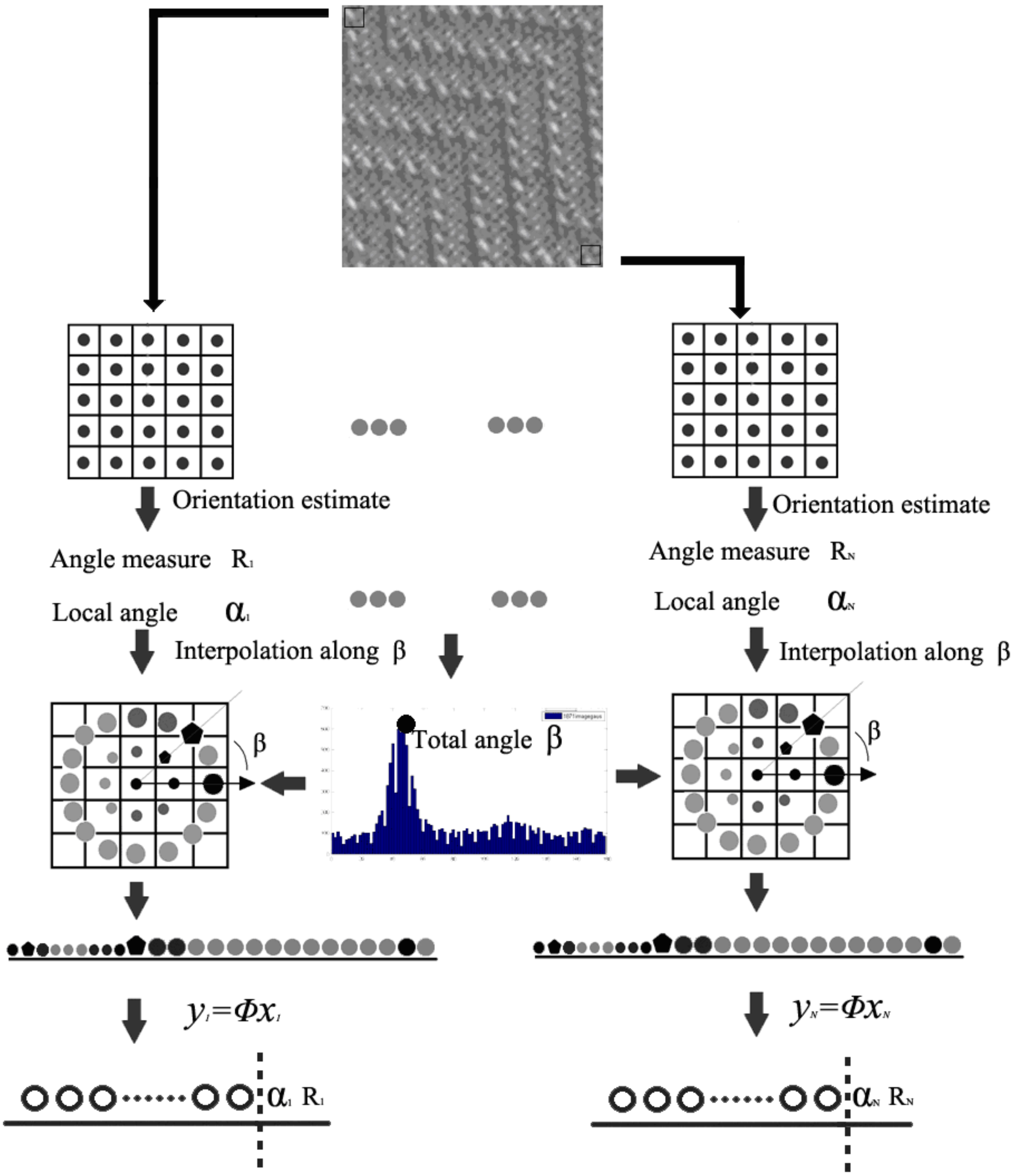


Figure 5. Feature extractions

(CUReT) database. Each intensity image is individually normalized to have an average intensity of zero and standard deviation. And other experimental setup is the same as in [10].

4.1. **Experimental Results on Outex Database.** The Outex database contains a large collection of textures with variations to illumination, rotation and spatial resolution. On this section, two test suites are chosen from the sixteen texture classification suites: the Outex_TC_00010(TC10) test suite and the Outex_TC_00012(TC12) test suite.

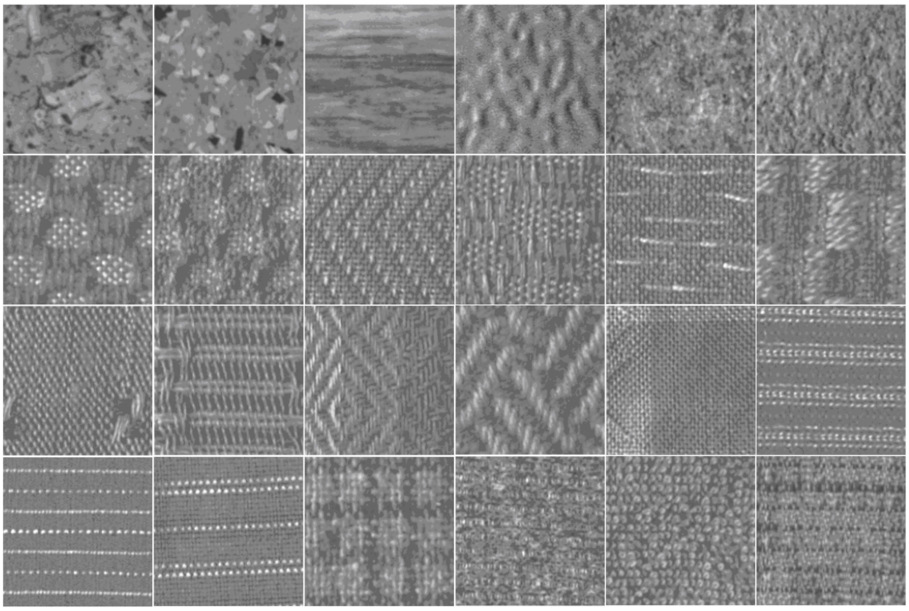


Figure 6. Selected Outex database textures

Both of them contain the same 24 texture classes at nine angles (0, 5, 10, 15, 30, 45, 60, 75, and 90) with 20 images per rotation angle. TC10 test suite contains 4320 Inca illumination texture images, and TC12 test suite contains 4320 Inca illumination texture images, 4320 horizon illumination texture images and 480 tl84 illumination texture images. As shown in table1, in the TC10 test suite, 24×20 images under Inca illuminant with rotation angle 0 are selected as the training samples, and other 24×160 images are selected as testing samples. For TC12 test suite, the training samples chosen are the same as that in TC10 test suite, and other $24 \times 180 + 480$ images under horizon and tl84 illuminant are used for test. Besides, all the training samples in experiments are used to get 40 textons per class and the low dimensional feature is one third of the high dimensional one.

Table 1. Experimental setup on Outex Texture database

database	class	Train angle (in °)	Test angle (in °)	Train samples	Test samples
TC10	24	0	5,10,15,30,45,60,75,90	480	3840
TC10	24	0	0,5,10,15,30,45,60,75,90	480	4800

Table 2. Experimental result on Outex database

	Pathc5 (in %)			Pathc7 (in %)		
	Tc10	Tc12t	Tc12h	Tc10	Tc12t	Tc12h
VZ_joint[10]	-	-	-	98.51	97.45	98.35
CLBC_CLBP[15]	98.83	93.59	94.26	98.96	95.37	94.72
Joint_sort[10]	-	-	-	99.19	94.88	96.82
OP(Proposed)	98.33	92.29	95.67	99.04	97.08	97.50

The experimental results on Outex dataset are listed in table2. The results of VZ_joint and Joint_sort method are copied from [10], and the results of CLBC_CLBP are copied from [15]. The classification accuracy rate of PO method is a litter lower (-0.15%) than joint_sort method in TC10 test suit, but much higher (+2.2%, +0.68%) than it in TC12 test suit. And the classification correct rate of PO method is higher (+0.53%) than VZ_joint method in TC10 test suit, while lower (-0.37%, -0.85%) than it in TC12 test suit. Besides if we use the 360circular interpolation in the feature extraction stage, the classification accuracy rate on TC10 test suit of PO method can achieve 99.13%. It can be seen that the proposed method has a better performance than Joint_sort on Outex database experiments. And they have a comparative time cost during the experiments by matlab2013. And the PO method achieves a little lower accuracy than VZ_joint on Outex database. The reason may be that it is hard to accurately estimate the angle of the textures sometimes.

4.2. **Experimental Results on CUReT Database.** The Columbia-Utrecht (CUReT) database contains images of 61 materials images under different viewing and illumination conditions. To compared with other methods, the same 92 images per classes are selected and a 200×200 central region is cropped from each images. Besides, all the images are converted to grey scale.

TABLE 3. Experimental result on Outex database

	L (in %)			
	46	23	12	6
VZ_joint[10]	97.510.75	94.271.63	89.002.26	80.223.93
Joint_sort[10]	96.930.95	93.001.92	86.283.11	76.244.16
PO	95.430.38	90.620.56	83.430.84	72.881.07
360PO	97.230.33	93.590.50	87.660.72	78.371.03
PO/NO-RP	97.280.34	93.350.52	87.270.72	77.941.03
360PO/NO-RP	96.950.35	92.930.51	86.660.76	77.211.03

In the experiments, the patch size n is 7, and the low dimensional feature is one third of the high dimensional one. L samples per class are chosen as the training samples. The first 23 images of each class are used for clustering to get 40 textons per class. And other $(92 - L) \times 61$ images are selected for test. Table2 shows the experiment results on CUReT database. To get the significant difference, the experiments are taken over a thousand random splits, and the accuracy represents the average classification rates and standard deviation. PO is the proposed method. The average accuracy of PO method (95.43%, 90.62%, 83.43%, 72.88%) is worse than Joint_sort (-1.50%, -2.74%, -2.85%, -3.36%) and VZ_joint (-2.08%, -3.65%, -5.57%, -7.34%) method. To get the reason, three improved algorithm of PO method is tested on the database. 360PO presents the method which using 360circular interpolation in the feature extraction stage. PO/NO-CS indicates the PO method without random projection stage. 360PO/NO-RP is the method using 360circular interpolation and without random projection. The average accuracy of PO/NO-CS method (97.28%, 93.35%, 87.27%, 77.94%) is higher than Joint_sort (+0.35%, +0.35%, +0.99%, +1.7%) and a little worse than VZ_joint (-0.23%, -0.92%, -1.73%, -2.28%) method. It is noted that the random projection can reduce the computational complexity, but also reduce the accuracy of algorithm at the same time in this method. The average accuracy of 360PO method (97.23%, 93.59%, 87.66%, 78.37%) is higher than Joint_sort (+0.30%,

+0.59%, +1.38%, +2.13%) and a little worse than VZ_joint (-0.28%, -0.68%, -1.34%, -1.85%) method. It can be seen that the finely complex interpolation can improve the performance of the PO method. While when we apply the random projection on the 360PO method, the average accuracy rises. And when we use the 360° circular interpolation on the PO/NO-CS method, the average accuracy falls. This phenomenon can be explained like this. No matter the finely complex interpolation or random projection, they have changed the information we get in the RITC. For a texture database, if we get enough information for RITC, redundant information causing by finely complex interpolation will disturb the classification. And if the information we have is suitable, reducing information by random projection will surely lead to the decrease of the accuracy. Besides standard deviation accuracy of the PO method and its improved method are lower than VZ_joint and Joint_sort method. It demonstrates that, on the CUReT database, the proposed methods have a stable RITC performance.

5. **Conclusions.** In the proposed method, the local orientation is combined with the local patch based method. The local orientation estimated by SVD of the gradient image patch is used in two ways, one is for getting the dominant orientation, and the other is gathered into the final feature vector. To reduce the computational complexity, random projection is applied to mapping the high dimension patch vector into a low dimension one. The experiment results show the proposed method has a better performance than joint_sort method on Outex database, and has a little worse result in CUReT database. Although the random projection can reduce the computational complexity, the classification accuracy rate is not satisfactory enough. More experiments should be taken to find how to accurately estimate the angle of the textures and how to keep the suitable information.

Acknowledgment. This work is partially supported by National Natural Science Foundation of China 61371175, Postdoctoral Science-research Developmental Foundation of Heilongjiang Province LBH-Q09128, and Fundamental Research Funds for the Central Universities HEUCFQ1411. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] T. Mäenpää, T. Ojala, M. Pietikäinen, and S. Maricor, Robust Texture Classification by Subsets of Local Binary Patterns, *Proc. 15th Int'l Conf. Pattern Recognition*, vol. 3, pp. 947-950, 2000.

[2] T. Ojala, M. Pietikäinen and T. Mäenpää, Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971-987, 2002.

[3] S. Arivazhagan, L. Ganesan and T. G. Subash Kumar, Texture classification using ridgelet transform, *Pattern Recognition Letters*, vol. 27, no. 16, pp. 1875-1883, 2006.

[4] W. Pan, T. D. Bui and C. Y. Suen, Rotation invariant texture classification by ridgelet transform and frequency-orientation space decomposition, *Signal Processing*, vol. 88, no. 1, pp.189-199, 2008.

[5] H. G. Kaganami, S. K. Ali and B. Zou, Optimal approach for texture analysis and classification based on wavelet transform and neural network, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, pp. 33-40, 2011.

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

[7] M. Varma and A. Zisserman, Texture classification: Are filter banks necessary, *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp.II691-II698, 2003.

[8] M. Varma and A. Zisserman, A statistical approach to material classification using image patch exemplars, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2032-2047, 2009.

[9] L. Liu and P. Fieguth, Texture classification from random features, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 574-586, 2012.

[10] Z. H. Guo, Q. Li, L. Zhang, J. You, D. Zhang and W. H. Liu, Is local dominant orientation necessary for the classification of rotation invariant texture, *Neurocomputing*, vol. 116, pp. 182-191, 2013.

[11] M. Varma, *Statistical Approaches to Texture Classication*, Ph.D. Thesis, Oxford University, Jesus College, 2008.

[12] X. G. Feng and P. Milanfar, Multiscale Principal Components Analysis for Image Local orientation estimation in *The 36th conference on Signals, Systems and Computers*, vol.1, pp.478-482, 2002.

[13] X. G. Feng, *Analysis and approaches to image local orientation estimation* Ph.D. Thesis, California Santa Cruz, 2003.

[14] Candès and J. Emmanuel, Compressive sampling. International Congress of Mathematicians, *ICM*, vol. 3, pp. 1433-1452, 2006.

[15] Y. Zhao, D. S. Huang and W. Jia, Completed Local Binary Count for Rotation Invariant Texture Classification, *IEEE Trans. on Image Processing* , vol. 21, no. 10, pp.4492-4497, 2012.