Texture Classification by Combining Wavelet and Contourlet Features

Shutao Li[1,2] and John Shawe-Taylor[2]

[1] College of Electrical and Information Engineering, Hunan University,
Changsha, 410082, P.R. China
shutao_li@yahoo.com.cn

[2] ISIS Research Group, School of Electronics and Computer Science,
University of Southampton, SO17 1BJ, UK
jst@ecs.soton.ac.uk

Abstract. In the recent decades, many features used to represent a texture were proposed. However, these features are always used exclusively. In this paper, a novel approach is presented that combines two types of features extracted by discrete wavelet transform and contourlet transform. Support vector machines (SVMs), which have demonstrated excellent performance in a variety of pattern recognition problems, are used as classifiers. The algorithm is tested on four different datasets, selected from Brodatz and VisTex database. The experimental results show that the combined features result in better classification rates than using only one type of those.

1 Introduction

Texture analysis is important in many applications, such as object recognition, image retrieval, remote sensing, and biomedical image analysis, scene interpretation and segmentation.

The method for feature extraction from texture is critical to the success of the texture classification. Many methods have been proposed to extract texture features, such as the co-occurrence matrices [1], the Markov random fields [2], fractals [3], and the Gabor filters [4], wavelet transforms [5,6] and quadrature mirror filters [7]. Recently, Randen and Husøy did an extensive review and comparative study for texture classification on most major filtering-based approaches [8]. They concluded that various filtering approaches yield different results for different images.

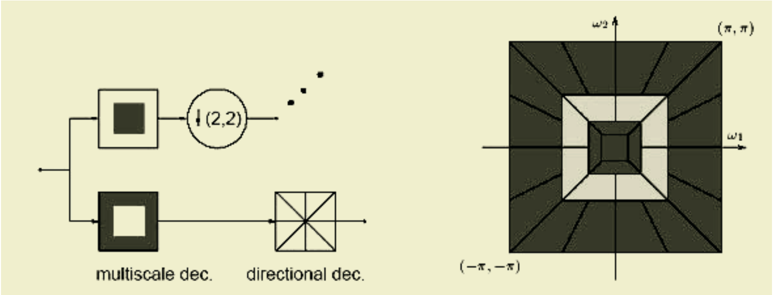
However, Most of these previous studies have used on the features individually. In this paper, a novel approach is presented that combines two types of features extracted by discrete wavelet transform and contourlet transform. Compared to wavelet transform, which can only offer limited directional information in representing image edges, the contourlet proposed by Do and Vetterli can capture the intrinsic geometrical structure in images [9]. Unlike other transforms, such as curvelets, that were initially developed in the continuous-domain and then discretized for sampled data, the contourlet starts with a discrete-domain transform. The discrete contourlet transform

A. Fred et al. (Eds.): SSPR&SPR 2004, LNCS 3138, pp. 1126–1134, 2004.

has a fast-iterated filter bank algorithm that requires an order N operation for N-pixel images.

The classifier is also clearly critical in texture classification. The support vector machine [10,11], which has found important applications in image classifications, is used as classifiers for texture classification.

This paper is structured as follows. Section 2 briefly introduces the contourlet. In section 3 the feature extraction methods using wavelet and contourlet are described. Section 4 describes the datasets, experiments and their results. Conclusions can be found in section 5.



(a) (b)

Fig. 1. (a) Block diagram of the contourlet filter bank. First, a standard multiscale decomposition into octave bands is computed, where the lowpass subband is subsampled and iterated, while a directional filter bank is applied the to the bandpass subband. (b) Resulting frequency division, where the number of directions is increased with frequency

2 Contourlet

Do and Vetterli developed the contourlet representation based on an efficient two-dimensional nonseparable filter banks that can deal effectively with images having smooth contours [9]. The block structure for the contourlet filter bank is shown in Figure 1 together with an example of its frequency partition. In contourlet transformation, the Laplacian pyramid is first used to capture the point discontinuities, then followed by a directional filter bank to link point discontinuities into linear structures. Contourlets possess not only the main features of wavelets, namely multiresolution and time-frequency localization, but they also show a high degree of directionality and anisotropy. Precisely, contourlet transform involves basis functions that are oriented at any power of two's number of directions with flexible aspect ratios. With such richness in the choice of bases, contourlets can represent any one-dimensional smooth edges with close to optimal efficiency.

3 Feature Extraction and Combination

The original texture sample is first decomposed using wavelet and contourlet. For each filtered subband image (except the one from the approximation subband) from wavelet, its mean and standard deviation are used to identify the textures based on the common belief. Denote the M×N image obtained in subband i by I_i, its mean and standard deviation are shown as follows.

Mean:

$$MEAN = \frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} |I_i(x,y)| \tag{1}$$

Standard deviation:

$$STD = \sqrt{\frac{1}{M \times N} \sum_{x=1}^{M} \sum_{y=1}^{N} (I_i(x,y) - MEAN)^2} \tag{2}$$

By decomposing the image to d levels using wavelet, we thus obtain a feature vector of length 6×d.

To contourlet, only the means of absolute decomposition values in subbands are used as feature vector. Figure 2 shows the contourlet representation of one texture image with [2 3] decomposition levels, from which we can extract 13 features.

Commonly, the exploitation of different information sources for the same recognition task often leads to different errors in the recognition results, which are very often complementary. This means that an appropriate exploitation of these sources can effectively reduce the error rate. Thus, we have chosen the strategy of using the combination of features from wavelet and contourlet for texture classification. Here the Combination of features from wavelet and contourlet are achieved by simply concatenating them.

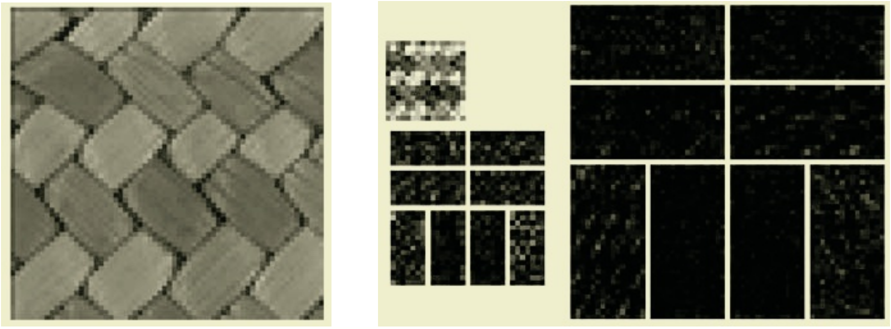

(a) (b)

Fig. 2. (a) One texture region; (b) Its contourlet transform

4 Experimental Results

4.1 Data

In this experiment, we have applied our method to four datasets from two commonly used texture sources: the Brodatz album and the MIT Vision Texture database.

The dataset 1 has 40 textures with size of 256×256, ten of which are shown in Fig.3. This dataset is challenging because there are significant variations within some textures and some of them are very similar to each other [12]. The dataset 2 and 3, shown in Fig. 4 (a) and (b), both have 10 textures are challenging too, which with size of 128×128, used in [8] and [12]. For these two groups, due to the inhomogeneity and large variations, texture types in local windows are perceptually close. The dataset 4 has 28 textures with size of 256×256 used in [5]. All textures are gray-scale images when presented to the methods. The dynamic ranges are represented by eight bits per sample.

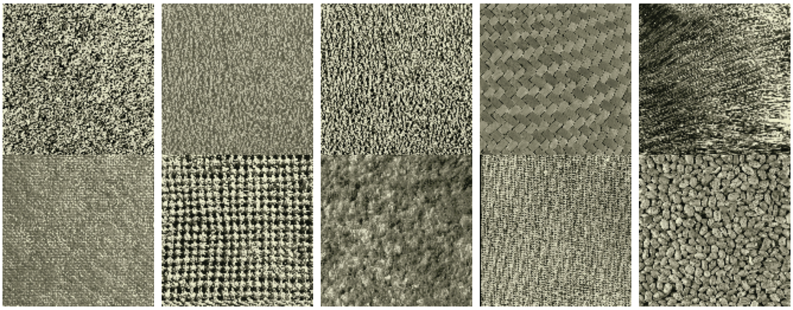
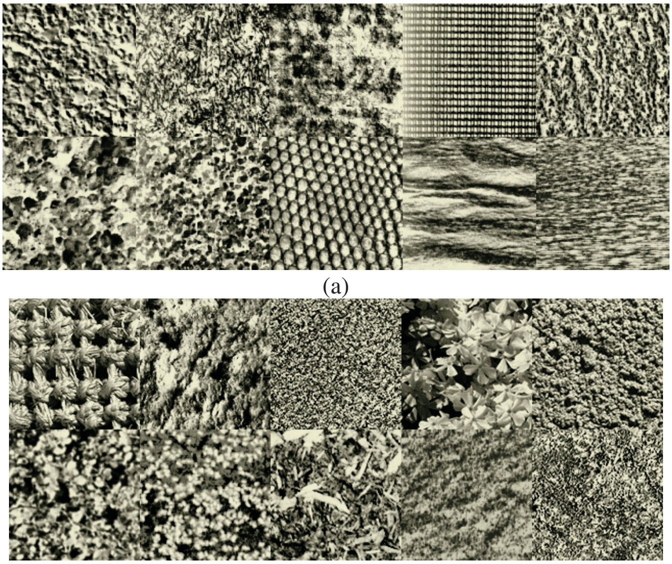


Fig. 3. Ten of the dataset 1 with 40 textures. The input image size is 256×256. These images are available at http://www-dbv.cs.uni-bonn.de/image/texture.tar.gz



(a)

(b)

Fig. 4. (a) Dataset 2 with 10 textures; (b) Dataset 3 with 10 textures. The image size is 128×128. These images are available at http://www.ux.his.no/~tranden/

In this experiment, we use a complete separation between the training and test sets and repeat the experiment 100 times and compute the average performance.

The classification gain defined by [12] is adopted as a measure to test the classification performance,

$$G = \frac{1 - C_{err}}{\frac{1}{M}} = M(1 - C_{err}) \tag{3}$$

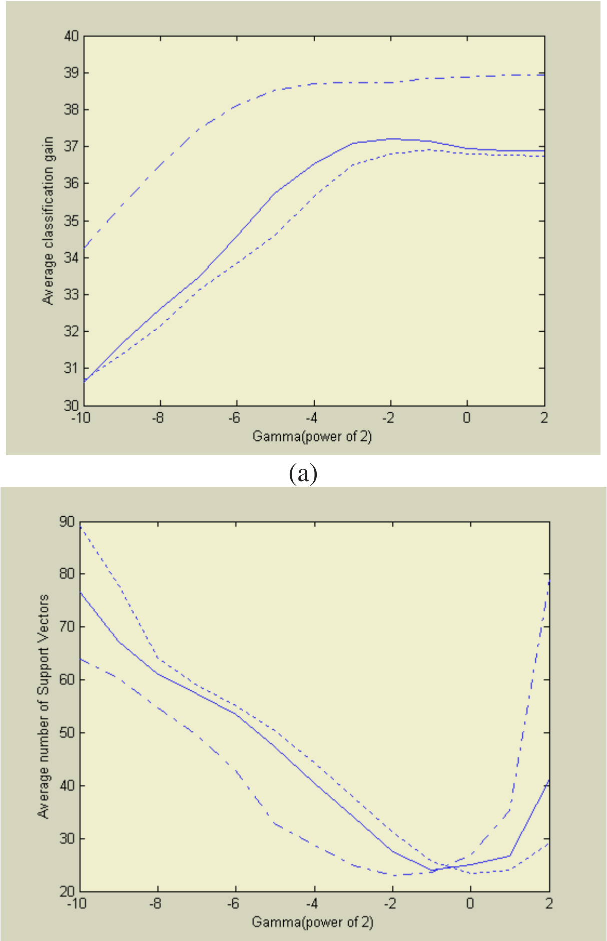
where C_{err} is the classification error rate, M is the total number of classes in the database. G measures the effectiveness of classifiers more objectively than C_{err} because is C_{err} closely related to M.

4.2 Selection of SVM Parameters

SVMs were originally designed for binary classification. Dealing with multi-class problem, SVMs can use one-against-the rest, one-against-one, error-correcting output coding and other methods. In this paper, we adopt the one-against-the rest. For training a class-C_i, the one-against-the rest method labels the data for the class-C_i as +1 and the data for other classes as –1. On classifying a new sample, the classifier with the largest output will be selected as the winner, and this new sample is assigned to the winner's corresponding texture class.

In the experiment, the Gaussian kernel will be used in the SVM, because preliminary results suggest that the Gaussian kernel outperforms the polynomial kernel. To select a suitable gamma value for Gaussian kernel function, we use the dataset 1, where size is 256×256, is subdivided into 64 nonoverlapping samples of size 32×32, resulting in a total of 2560 samples. For each sample, 3 level wavelet using “coif4” filters and [2 3] level contourlet with “pkva” filters are separately applied, so 18 and 13 features are obtained, respectively, resulting in the combined feature vector with length of 31. In the total samples, 50% are used for training and the rest of 50% are used for testing. The figures of average classification gain and the average number of support vectors versus gamma value are shown in Fig.5(a) and Fig.5(b), respectively, where C value is set to 1000.

It can be found that the classification gain on the combined feature is much better than those on wavelet feature and contourlet feature individually from Fig.5(a). The classification gains using individual wavelet features and contourlet features reach the maximum when gamma equals to 2^{-3} and 2^{-2}, respectively. And the classification gain using the combined features is increasing when the gamma value increases. But when gamma be larger than 2^{-5}, the increasing trend is slow. As shown in Fig.5(b), to the classifiers using wavelet feature and contourlet feature individually, the average number of support vectors is the fewest when the gamma value is about 0.5 and 1, respectively. The classifiers with combined features have fewest average number of support vectors when gamma equals to 0.25. After gamma be larger than 2, the average number of support vectors are increasing dramatically. So during the following experiments, considering the relationship between classification gain and computation complexity, the gamma is set to 1 and C value is 1000.



(a)

(b)

Fig. 5. (a) Average classification gain versus gamma (b) Average number of SVs versus gamma. Solid line-wavelet based; Dotted line-contourlet based; Dash-dot line-combined features based

4.3 Experimental Results and Comparisons

Table 1 shows the classification gain on dataset 1 using window size of 32×32. The experimental setup is same to the previous section. Different proportions of samples for training are applied to investigate the generalization capability of the proposed method. From the table, we can see that the classification result by using the combined features does not change too much for the training samples ranging from 12.5% to 75% of the total samples. The best result in [12] on that dataset is given in Table 1 for comparison. From the result we can see that, for the case 50% for training and 50% for testing, the classification gains of individual wavelet and contourlet features are slightly worse than that of the spectral histogram-based method, but the classifiers using combination of these two features can obtain much better results.

Table 1. Average classification gains of the proposed method on dataset 1 with window size of 32×32

Ratio	Wavelet	Contourlet	Combined	Method in [12] (best)
12.5%	33.95	33.62	37.27	N/A
25%	35.82	35.54	38.27	N/A
50%	**37.00**	**36.81**	**38.87**	**>=37**
75%	37.41	36.42	39.13	N/A

In order to analyze the effect of the window size, samples of size 64×64 pixels are also be used for experiment. So the original image is subdivided into 16 nonoverlapping windows, resulting in a total of 640 samples. The classification results are shown in Table 2. It can be seen the result is also very good.

Table 2. Average classification gains of the proposed method on dataset 1 with window size of 64×64

Ratio	Wavelet	Contourlet	Combined
12.5%	33.13	34.89	36.74
25%	35.85	37.09	38.30
50%	38.18	38.62	39.44
75%	38.77	38.94	39.57

The experimental results on the dataset 2 and 3 are given in Table 3 and Table 4, respectively. The original image is with size of 128×128, which is divided into 16 32×32 regions. The best results of the methods in [8] and [12] are also given in the tables. As we can see, in the case of 50% samples for training and 50% for testing, the proposed method can get significantly improvement to those methods.

Table 3. Average classification gains of the proposed method on dataset 2

Ratio	Wavelet	Contourlet	Combined	Method in [8] (best)	Method in [12] (best)
12.5%	8.04	7.31	8.51	N/A	N/A
25%	8.53	8.23	9.14	N/A	N/A
50%	**9.01**	**8.76**	**9.52**	**6.77**	**8.31**
75%	9.28	9.09	9.72	N/A	N/A

To further illustrate our method, we have done a comparison with a method proposed by [5], which on the dataset 4. The original image, where size is 256×256, is subdivided into 64 nonoverlapping samples of size 32×32, resulting in a total of

1792 samples. Eighty percent of samples are used for learning and the rest are used for testing the classifier. The classification performance is evaluated using five different randomly selected learning and testing sets. The results shown in Table 5, from which we can see that the proposed method can get better results than the method in [5].

Table 4. Average classification gains of the proposed method on dataset 3

Ratio	Wavelet	Contourlet	Combined	Method in [8] (best)	Method in [12] (best)
12.5%	7.19	6.62	7.37	N/A	N/A
25%	7.88	7.08	7.90	N/A	N/A
50%	**8.33**	**7.66**	**8.58**	**7.22**	**7.91**
75%	8.78	7.87	8.92	N/A	N/A

Table 5. Average classification gains of the proposed method on dataset 4

Wavelet	Contourlet	Combined	Method in [5] (best)
27.42	**27.50**	**27.88**	**27.36**

5 Conclusions

In this paper, we have proposed a method for texture classification by using combined features from wavelet and contourlet. SVMs with Gaussian kernel are used as classifiers. Experiments are performed on four different texture datasets selected from Brodatz and VisTex database. Experimental results demonstrated the combination of the two feature sets always outperformed each method individually. Comparative results to other methods show that the proposed method can get better results than other methods.

References

1. Gibson, J.J.: The Perception of The Visual World. Riverside Press, Cambridge (1950)
2. Cesmeli, E. and Wang, D. L.: Texture Segmentation using Gaussian-Markov Random Fields and Neural Oscillator Networks. IEEE Trans. Neural Networks 12 (2001) 394–404
3. Keller, J.M., Chen, S., and Crownover, R.M.: Texture Description and Segmentation through Fractal Geometry. Computer Vision and Graphical Image Processing 45 (1989) 150–166
4. Jain, A.K. and Farrokhnia, F.: Unsupervised Texture Segmentation using Gabor Filters. Pattern Recognition 24 (1991) 1167–1186
5. Kim, N.-D. and Udpa, S.: Texture Classification using Rotated Wavelet Filters. IEEE Trans System, Man, and Cybernetics-PART A: Systems and Humans 30 (2000) 847–852

6. Arivazhagan, S. and Ganesan, L.: Texture Classification using Wavelet Transform. Pattern Recognition Letters 24 (2003) 1513–1521
7. Randen, T. and Husøy, J.H.: Multichannel Filtering for Image Texture Segmentation. Optical Engineering 33 (1994) 2617–2625
8. Randen, T. and Husøy, J.H.: Filtering for Texture Classification: A Comparative Study. IEEE Trans Pattern Analysis and Machine Intelligence 21 (1999) 291–310
9. Do, M. N. and Vetterli, M.: Contourlets: A Directional Multiresolution Image Representation. Proc. of IEEE International Conference on Image Processing, Rochester, September 2002.
10. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
11. Cristianini, N. and Shawe-Taylor, J.: An Introduction to Support Vector Machines. Cambridge University Press, Cambridge (2000)
12. Liu, X.W., and Wang, D.L.: Texture Classification using Spectral Histograms. IEEE Trans Image Processing 12 (2003) 661–670