Texture Classification Using Cyclic Spectral Function

Mehdi Chehel Amirani  
Ali Asghar Beheshti Shirazi  
Electrical Engineering Department  
Iran University of Science and Technology  
Tehran, Iran  
amirani@ee.iust.ac.ir  
abeheshti@iust.ac.ir

Abstract

In this paper, a new feature extraction technique for texture classification is proposed. Features are energy and standard deviation of spectral correlation function (SCF) of signals got from image at different regions of bifrequency plane. This scheme shows high performance in the classification of Brodatz texture images. Experimental results indicate that the proposed method improves correct classification rate in comparing with traditional discrete wavelet transform approaches.

1. Introduction

Texture is a fundamental characteristic in many natural images and also plays an important role in machine vision and pattern recognition. Texture analysis is an essential step for many image processing applications such as industrial inspection, document segmentation, remote sensing of earth resources, and medical imaging. Texture analysis methodology has been investigated by a number of researchers. Recently, wavelet transform and wavelet packet are extensively used to extract feature vectors of texture images [1]-[3].

In this paper a novel texture feature extractor based on cyclostationary analysis is introduced. Cyclostationary waveforms are persistent random waveforms with statistical parameters that vary periodically with time. The spectral correlation function, which is the cross spectral of a signal and a frequency shifted version of itself, provides a second-order statistical description in the frequency domain of such signals. The theory and utility of cyclostationary signal models and the SCF, also called the cyclic spectrum, are discussed in references [4]-[9]. The basic time-smoothing and frequency-smoothing methods of spectral correlation analysis were introduced in [4] and proof of their equivalence was given in [5] and [6].

Methods which more fully exploit the computational efficiency of the FFT, namely, the FFT accumulation method (FAM) and the strip spectral correlation analyzer (SSCA) were introduced in [7] and discussed in [10] and [11]. Modulation classification based on cyclic spectral analysis has a good performance [12] and we show that this idea can be applied to feature extraction for texture classification.

The paper is organized as follows. We discuss the spectral correlation function in section 2. In section 3, the concept of texture classification based on SCF is explained. Section 4 presents experimental results. Finally, the conclusion is given in section 5.

2. Spectral Correlation Function

The SCF for a discrete-time real-valued signal \( x(n) \) is defined as the Fourier-series transform of the cyclic correlation function \( R_x^\alpha (k) \),

\[
S_x^\alpha (f) = \sum_{k=\infty}^{\infty} R_x^\alpha (k) e^{-j2\pi fk}
\]

where

\[
R_x^\alpha (k) = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{n=-N}^{N} [x(n+k)e^{-j\pi\alpha(n+k)}][x(n)e^{j\pi\alpha n}]^*
\]

Thus \( S_x^\alpha (f) \) is the cross spectrum of the pair of complex valued frequency-shifted signals \( x(n)e^{-j\pi\alpha n} \) and \( x(n)e^{j\pi\alpha n} \), where \( f \) is the cross spectrum frequency variable and the parameter \( \alpha \), called the cyclic frequency, is the relative frequency shift. If the signal has finite average power then there are at most a countable number of values of \( \alpha \) for which
Mathematically, computation of the complex demodulates is expressed as:

$$X_T(n, f) = \sum_{n=-N/2}^{N/2} a(r)x(n-r)e^{-j2\pi f(n-r)T_s}$$

where $a(r)$ is a data tapering window of length $T = N'T_s$. Then, the complex demodulates are correlated over a time span of $\Delta t$ seconds:

$$S_{X_T}(n, f_0)_{\Delta t} = \sum_{r} X_T(r, f_1)X_T^*(r, f_2)g(n-r)$$

where $g(n)$ is a data tapering window of width $\Delta t = NT_s$; $f_1 = f_0 + \alpha_0/2$ and $f_2 = f_0 - \alpha_0/2$. It is shown in [13] that the time smoothed cyclic cross periodogram converges to the cyclic cross spectrum in the limit, as $\Delta t \to \infty$ followed by $\Delta f \to 0$, if the time windows $a(n)$ and $g(n)$ are properly normalized. Therefore, if $\sum n^2 = \sum g(n) = 1$, then

$$\lim_{\Delta f \to 0, \Delta t \to \infty} S_{X_T}(n, f_0)_{\Delta t} = S_{X_T}^0(f_0)$$

In the FAM algorithm, time smoothing is done by Fourier transform. If frequency is shifted from $\alpha_0$ to $\alpha_0 + \epsilon$, the output of the system is given by:

$$S_{X_T}^{\alpha_0+\epsilon}(n, f_0)_{\Delta t} = \sum_{r} X_T(r, f_1)X_T^*(r, f_2)g(n-r)e^{-j2\pi \epsilon \alpha_T}$$

If several values of $\epsilon$ are desired, evaluation of the sum can be simplified by discretizing the values of $\epsilon$ to be $\epsilon = q\Delta f$. In this case the output of the algorithm is expressed as:

$$S_{X_T}^{\alpha_0+q\Delta f}(n, f_0)_{\Delta t} = \sum_{r} X_T(r, f_1)X_T^*(r, f_2)g(n-r)e^{-j2\pi q\epsilon \alpha_T/N}$$

In which the sum can be evaluated with an N-point FFT. Thus, point estimates with constant cycle frequency can be computed in blocks by Fourier transforming the product sequences instead of averaging the product sequences individually.

For complete coverage of the bifrequency plane a bank of bandpass filters is required to produce the necessary complex demodulates. An efficient method for producing the required complex demodulates is based on a sliding FFT [13]. In this approach the frequencies of the filter bank are discretized to:

$$f_k = k(f_1/N'), k = -N'/2 \ldots (N'/2) - 1$$

835
The locations of SCF estimation associated with the pairs of complex demodulates are \( (f_j, \alpha_i) \) where the frequency coordinates are:

\[
f_j = \frac{f_k + f_i}{2} = \frac{k + l}{2} \left( \frac{f_s}{N'} \right)
\]

And the cycle frequency coordinates are:

\[
\alpha_i = f_k - f_i = (k - l) \left( \frac{f_s}{N'} \right)
\]

Figure 1 shows tiling the bifrequency plane with the FFT accumulation method for \( N' = 8 \) and the locations of SCF estimation. For an \( N' \)-point channelizer there are \((N')^2\) possible combinations of channelizer streams; hence, there are at most \((N')^2\) estimation regions (diamond regions). Due to symmetry, estimation of the cyclic spectrum of a single real signal requires only \((N')^2 / 4\) diamond regions (one quadrant of the bifrequency plane). The more details of the FAM algorithm can be studied at [10].

3. The Proposed Texture Classification

In this section the proposed image classification method is explained. At first features are extracted from SCF and then these features are used to texture classification.

3.1. Texture Feature Extraction

Two one dimensional signals are obtained from each image from database by ordering of pixels row by row and column by column. Then the SCF of each signal is calculated by FAM algorithm. Figure 2 shows the Brodatz texture image D20 and its corresponded SCFs magnitudes for \( N' = 8 \). For proper visualization, at each cell the SCF has been extended at a square region.

For constructing the feature vector, the energy and standard deviation are computed separately on each region of SCFs from each signal and then feature vector is formed using these two parameter values. The basic assumption of using energy as a feature for texture discrimination is that the energy distribution in the frequency-domain identifies a texture. Resulting feature vectors from \( N \) number of regions from each signal are as follows.

\[
\vec{f} = \left\{ E_{11}, \ldots, E_{1N}, E_{21}, \ldots, E_{2N}, \sigma_{11}, \ldots, \sigma_{1N}, \sigma_{21}, \ldots, \sigma_{2N} \right\}
\]

where \( E_{in}, i \in \{1, 2\}, n = 1, 2, \ldots, N \) and \( \sigma_{in}, i \in \{1, 2\}, n = 1, 2, \ldots, N \) are the energy and standard deviation of the SCF for the \( i \)th signal at the \( n \)th region, respectively, and \( N \) is the number of diamond regions at FAM algorithm. The length of the feature vector will be equal to \( M = 4N \). Since image is a real signal, entire function is determined by \( S^a_f(f) \) for \( 0 \leq f \leq 1/2, 0 \leq \alpha \leq 1 - 2f \), so one quarter of regions makes a complete estimation. For the creation of feature database, the above procedure is repeated for all the images on the database and these feature vectors are stored in the feature database.
3.2. Texture Classification

We use a 3-layer multi layer perceptron (MLP) for the image classification by using the extracted features from the SCF (Figure 3). The first step of using a neural network is the training phase in which the back-propagation algorithm is used. In implementation of such algorithm, two distinct passes of computation may be distinguished. The former is referred to as the forward pass, and the latter as the backward pass. In the forward pass, the synaptic weights remain unaltered throughout the network, and the function signals of the network are computed on a neuron-by-neuron basis. The Backward pass, on the other hand, starts at the output layer by passing the error signals leftward through the network, layer by layer. Weights and biases are updated iteratively until the mean square error (MSE) is minimized.

Figure 3. Architecture of a MLP.

4. Experimental Results

We used 12 textures (D9, D11, D12, D17, D20, D24, D29, D53, D68, D84, D92, and D112) obtained from Brodatz’s texture album as shown in Figure 4 [14]. Each image is of size 512×512 pixels with 256 gray levels. A database of 768 image regions of 12 texture classes was constructed by dividing each 512 × 512 image into 64 nonoverlapping 64 × 64 regions. We have used 43 images from each class randomly for training of neural network and remained 21 images from each class to test the classifier.

We compare the result of our method with wavelet based feature extraction methods consist of Gabor, standard real discrete wavelet transform (DWT), dual-tree-complex wavelet transform (DT-CWT) and dual-tree rotated complex wavelet filter (DT-RCWF). To have a fair comparison, the number of neurons at hidden layers at MLP is equal to 20 for all of classifiers.
The performance of classifier is evaluated in terms of the correct classification rate (CCR) at each class and average CCR for all 12 classes. Table 1 shows the CCR for test images. The percentage of average CCR is 98.4127% by Gabor features, 95.6349% by DT-CWT features, 96.4286% by DT-RCWF features, 93.6508% by DWT features and 99.6031% by SCF features. These results indicate that our method improves classification performance.

5. CONCLUSION

We have introduced a new texture classification based on cyclic spectral analyzer. Two spectral correlation functions are calculated with a computationally efficient algorithm named FAM. The sum and standard deviation of SCF magnitude at each cell obtained from FAM algorithm have been used as texture features. The proposed method showed high performance in the classification of the Brodatz texture images.

Table 1. The percentage of CCR for test images (%)

<table>
<thead>
<tr>
<th></th>
<th>D19</th>
<th>D11</th>
<th>D12</th>
<th>D17</th>
<th>D20</th>
<th>D24</th>
<th>D29</th>
<th>D33</th>
<th>D68</th>
<th>D84</th>
<th>D92</th>
<th>D112</th>
<th>Average CCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCF</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>95.24</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>99.6032</td>
</tr>
<tr>
<td>Gabor</td>
<td>95.24</td>
<td>100</td>
<td>90.48</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>98.4127</td>
</tr>
<tr>
<td>DT_CWT</td>
<td>95.24</td>
<td>100</td>
<td>90.48</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>85.71</td>
<td>76.19</td>
<td>95.6349</td>
</tr>
<tr>
<td>DT_RCWF</td>
<td>85.71</td>
<td>100</td>
<td>76.19</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>96.4286</td>
</tr>
<tr>
<td>DWT</td>
<td>80.95</td>
<td>95.24</td>
<td>85.71</td>
<td>100</td>
<td>100</td>
<td>90.48</td>
<td>85.71</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>95.24</td>
<td>90.48</td>
<td>93.6508</td>
</tr>
</tbody>
</table>

6. REFERENCES