

**Title of the Paper:**

**Unsupervised segmentation of texture images using a  
combination of Gabor and wavelet features**

**Authors:**

**Shivani G. Rao<sup>\$</sup>, Manika Puri\*, Sukhendu Das<sup>\$</sup>**

**Address:**

**<sup>\$</sup>Dept. of Computer Science and Engineering,  
<sup>\*</sup>Dept. of Applied Mechanics.  
Indian Institute of Technology, Madras,  
Chennai- 600036**

**Email :**

**(shivani@peacock., am03s015@violet., sdas@)iitm.ernet.in**

**Address of correspondence:**

**Dr. Sukhendu Das  
Dept. of Computer Science and Engineering,  
Indian Institute of Technology, Madras,  
Chennai- 600036.**

**Email : sdas@iitm.ernet.in**

**Phone : (91)-44-22578346**

**Fax : (91)-44-22578352.**

# Unsupervised segmentation of texture images using a combination of Gabor and wavelet features

Shivani G. Rao  
Dept. of Computer Science and  
Engineering, IIT Madras,  
Chennai-600036  
shivani@peacock.iitm.ernet.in

Manika Puri  
Dept. of Applied Mechanics  
IIT Madras,  
Chennai-600036  
am03s015@violet.iitm.ernet.in

Sukhendu Das  
Dept. of Computer Science and  
Engineering, IIT Madras,  
Chennai-600036  
sdas@iitm.ernet.in

## Abstract

*This paper presents an unsupervised method of texture classification by combining the two most commonly used multi-resolution, multi-channel filters: Gabor filters and wavelet transform. We used a set of 8 Gabor filters and 2 wavelet filters: Daubechies and Haar, for our analysis. The parameters (viz. frequency, orientation and size) of the Gabor filter bank are obtained by trial and error method, based on visual observation of an energy measure of the response. A fuzzy classifier has been used which uses no a priori knowledge of the textures and hence provides unsupervised segmentation. For comparing the performance of the features from the Gabor filter bank, the two wavelet filters separately and a combination of all the three, the classification algorithm was kept identical. A combination of Gabor and wavelet features provides better performance compared to the individual features alone.*

## 1. Introduction

In the field of computer vision, texture plays an important role in low-level image analysis and understanding. Its range of potential applications include analysis of remote sensing images, industrial monitoring of product quality, medical imaging, and recently, content-based image and video retrieval. There is no formal or unique definition of texture, making texture analysis a difficult and challenging problem. Classification and segmentation of texture content in digital images has received considerable attention during the past decades and numerous approaches have been presented. Statistical, model-based, and signal processing techniques are the most commonly used approaches.

The focus of this paper will be on multi-rate and multi-resolution signal processing approaches. A common denominator for most signal processing approaches is that the textured image is submitted to a linear transform, filter, or filter bank, followed by some energy measure. Two filtering-based texture feature extraction schemes have been presented. The focus will be on filtering, keeping the other components same. Texture segmentation deals with identification of regions where distinct textures exist, so that further analysis can be done on the respective texture regions alone. We have used the fuzzy c-means (FCM) classifier, which provides an unsupervised segmentation. In this paper we have used multi-texture images of size 256 X 256 having four or five distinct texture regions.

## 2. Brief review of related literature

Most researchers have attempted to use well-established and standard texture segmentation techniques for the identification of different texture surfaces. Most methods are based on wavelet features, MRF models, STFT features, co-occurrence matrices, geometric shape of texels and PCA analysis. We discuss here a few papers, relevant to our work, which deal with the segmentation of textures using wavelet transform and Gabor filters.

Dunn et. al. [2] presents an algorithm to design specially tuned Gabor filters to segment images with bipartite textures. The parameter tuning of the set of Gabor filter bank is the key contribution of this approach. Results are shown mostly on simulated and a few real world samples. Grigorescu et. al. [3] presents a comparative study of the different texture features based on Gabor filter bank outputs. The three features (obtained by non-linear processing) being compared are: Gabor energy, complex moments and grating cell operator features. Yegnanarayana et. al. [10] uses a pair of 1-D Gabor filters in orthogonal directions to process a texture image and obtain the texture boundaries quite accurately. Although the size of the filter bank is large, the efficiency of 1-D processing helps in reducing the computation complexity. Segmentation is edge based for this method.

Wavelet based methods have also been popular for texture segmentation. Charalampidis and Kasparis [1] use a set of new roughness features for texture segmentation and classification. Wavelets are used to extract single-scale and multiple-scale texture roughness features. These are then transformed to a rotational invariant feature vector, which has the information of texture direction. Iterative K-means algorithm has been used for segmentation and Baye's classifier for classification. Results are shown using a large set of real world texture images. Salari and Ling [8] used a hierarchical wavelet decomposition technique for texture image segmentation. Daubechies 4-tap filters were used to decompose the original image into three detail and one approximate sub-band images. A K-means clustering algorithm was used for segmentation of the image using textural features obtained from the different bands starting from the lowest band, where coarse resolutions provide information about larger structures and fine resolution provides the details for refining the results. Results are shown on a few regular and homogeneous real-world textures. Lu et. al. [4] proposed a method of unsupervised texture description using wavelet transform. The proposed methodology has four stages. The first stage computes a

smoothed local energy of the wavelet coefficients in high-frequency bands, as features for segmentation. The second stage performs a coarse segmentation using a multi-thresholding technique. In the third stage, the features at different orientations and scales are fused in intra-scale and inter-scale respectively. In the last stage, ambiguously labeled pixels are reclassified in a fine segmentation technique. Segmentation results at various scales are integrated by inter-scale fusion to determine the number of classes. Results are shown on a few real-world images, with the use of various types of wavelet filters. Mallet et. al. in [5] proposed a method to design adaptive wavelets for the purpose of classification of mineralogical spectral data. The purpose of designing adaptive wavelets was to optimize a specified discriminant criterion and reduce the dimensionality of the feature space. The choice of adaptive wavelets proved to be beneficial compared to standard wavelets such as Daubechies or Coiflet families - the classification accuracy was better.

Randon and Hüsoy [7] provide a comparative study of various types of filters (heuristically designed and optimized filter banks) for texture classification, which includes, Gabor dyadic filters, wavelet transforms, DCT, AR models, co-occurrence matrices and eigenfilters. They compare the filters, using classification errors and computational complexity as the performance criteria. One important result is that wavelets performed better than Gabor filters, in general. Pichler et. al. in [6] compares the pyramidal and tree-structured wavelet with the adaptive Gabor filtering for texture classification. Results show that both the wavelet-based methods are sub-optimal for feature extraction purpose, because the center frequency, orientation and bandwidth cannot be selected. The paper also concludes that Gabor filtering outperforms the wavelet cases but is computationally more expensive.

The work presented in our paper uses a combined representation of texture classification, based on Gabor and wavelet features. This representation combines the discriminability of these multi-rate, multi-resolution filters to provide improved segmentation results.

### 3. Overall Methodology

The steps of the overall methodology for texture classification are shown in Figure 1. The filtering stage consists of either a bank of Gabor filters or dyadic discrete wavelet transforms. The filter coefficients (responses) are post-processed using a set of non-linear functions, which compute the local energy estimates of the filtered coefficients. These non-linear functions consist of two stages: (i) subtracting the local mean and obtaining the magnitude followed by (ii) smoothing by a large Gaussian function. These steps are described in the section 3.3. The feature vectors computed from the local energy measure estimates are local mean and local variance, which represent local texture characteristics. These feature vectors are computed from the various filtered images and provided to the FCM to segment the texture patterns in the image. In our experiment, user provides the desired number of classes as an input to the classifier.

### 3.1 Gabor Filter Definition

The input image  $i(x, y)$  is comprised of disjoint regions of  $N$  textures  $t_1, t_2, \dots, t_N$  with  $N \geq 2$ . This input is applied to  $k$  filter channels, where each channel consists of a bandpass Gabor function  $h_j(x, y)$ . The Gabor filter in channel  $j$  has impulse response  $h_j(x, y)$ :

$$h_j(x, y) = \frac{1}{2\pi\sigma_{1j}^2} e^{-\frac{-(x^2+y^2)}{2\sigma_{1j}^2}} e^{-j2\pi(u_jx+v_jy)}$$

where  $(u_j, v_j)$  is the center frequency of the filter.

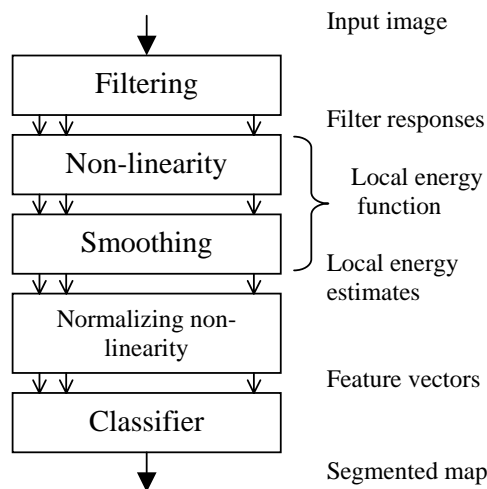


Figure 1: Stages of processing for texture classification

The filtered output  $ih_j(x, y)$  is the convolution of the input image with Gabor filter given by:

$$ih_j(x, y) = h_j(x, y) ** i(x, y)$$

where  $**$  denotes convolution in 2-D. Thus, the parameters  $(\sigma_{1j}, u_j, v_j)$  determine the response of filter channel  $j$ , and are varied to generate a bank of 2-D Gabor filters. Figure 2(a) and (b) show two typical examples of Gabor filters with different parameters. The corresponding filtered outputs of the image in Figure 3(c), are shown in Figure 2(c) and (d) respectively.

### 3.2 Discrete wavelet transform (DWT)

The discrete wavelet transform analyses a signal based on its content in different frequency ranges. Therefore it is very useful in analyzing repetitive patterns such as texture. The wavelet transform is expressed as a decomposition of a signal  $f(x) \in L^2(\mathcal{R})$  into a family of functions which are translations and dilations of a mother wavelet function  $\Psi(x)$ . Employing the definition:

$$\psi_x(x) = \sqrt{s}\psi(s(x-a))$$

the wavelet transform of  $f(x)$  is defined by:

$$Wf(s, a) = \int_{-\infty}^{+\infty} f(x)\sqrt{s}\psi(s(x-a))dx \quad (1)$$

where  $s, a \in \mathcal{R}$  indicate scale and translation parameters respectively. Since the continuous wavelet transform is redundant, it is discretized by sampling parameters,  $s, a$ .

The most common choice is  $s = 2^i$  and  $a = n/2^i$ ;  $i, n \in \mathbf{Z}$ . Inserting these values in equation (1) yields the DWT of the signal  $f(x)$  as:

$$W_d f(i, n) = \langle f(x), \psi_2(x - n2^{-i}) \rangle \quad (2)$$

where  $\langle \dots \rangle$  denotes the inner product.

Since some existing wavelets  $\Psi(x) \in L^2(\mathcal{R})$  constitute an orthonormal basis:

$$\{\psi_{2^i}(x - (n/2^i))\}; i, n \in \mathbf{Z},$$

this transform is called an orthogonal wavelet transform.

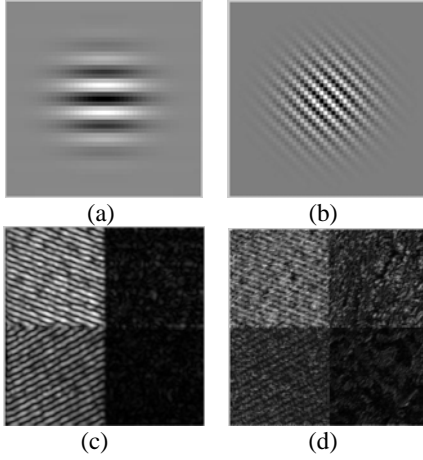


Figure 2: Real part of Gabor Filter at different scales ( $\sigma$ ) and orientations ( $\theta$ ): (a)  $\sigma=6, \theta=0^\circ, \omega=2$  (b)  $\sigma=6, \theta=45^\circ, \omega=2.8$ . (c), (d): Gabor filter responses for the sample image shown in Figure 3 (c)

Introducing the so-called scaling function  $\phi(x)$ , the interscale coefficients  $g(n)$  with high-pass (HP) characteristics and  $h(n)$  with low-pass (LP) characteristics, it is possible to decompose the signal  $f(x)$  using the following  $L$ -level decomposition scheme:

$$\begin{aligned} f(x) &= \sum_n c_{0,n} \phi(x-n) \\ &= \sum_{n=-\infty}^{\infty} c_{L,n} \phi_{2^{-L}}(x-n2^L) \\ &+ \sum_{i=1}^L \sum_{n=-\infty}^{\infty} d_{i,n} \psi_{2^{-i}}(x-n2^i) \end{aligned}$$

which is a finite approximation of equation (2). The coefficients  $c_{i,n}$  and  $d_{i,n}$  are obtained by

$$c_{i,n} = \sum_k c_{i-1,n} h(k-2n)$$

$$d_{i,n} = \sum_k c_{i-1,n} g(k-2n)$$

which is same as convolving the signal  $c_{i-1,n}$  with impulse responses

$$\tilde{h}(n) = h(-n), \quad \tilde{g}(n) = g(-n)$$

respectively and subsequently discarding every other sample. The 2-D transform uses a family of wavelet functions and its associated scaling function to decompose

the original image into different channels, namely the *low-low*, *low-high*, *high-low* and *high-high* ( $A, V, H, D$  respectively) channels. The decomposition process can be recursively applied to the low frequency channel ( $LL$ ) to generate decomposition at the next level. Figure 3(a), (b) show the 2-channel level-2 dyadic decomposition of an image. The LP and HP *filters* are used to implement the wavelet transform. This results in an output with the same size as that of the input. Figure 3(d) shows a level-1 DWT decomposition of a sample image in Figure 3(c).

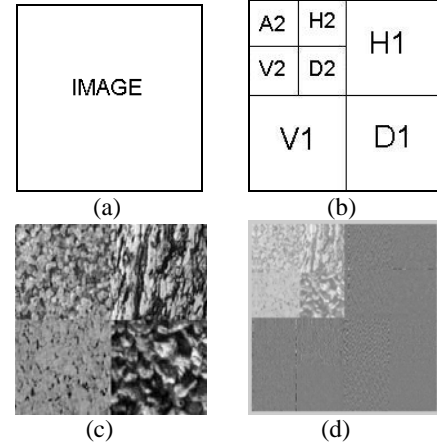


Figure 3: (a) Input image (b) Decomposition at level-2 (c) A texture image (d) The DWT level-1 coefficients using the Daubechies 8-tap filter

### 3.3 Post-processing of filter response

The filter coefficients are post-processed using a set of non-linear functions, which compute the local energy estimates (as shown in Figure 1). The mean-subtracted magnitude of the filter output is taken as:

$$m_j(x,y) = |ih_j(x,y) - \mu_j(x,y)|$$

where  $ih_j(x,y)$  is the  $j^{\text{th}}$  channel output of the filter and  $\mu_j(x,y)$  is the local mean image of the filter output. A lowpass Gaussian post-filter  $g_p(x,y)$  is then applied to each  $m_j(x,y)$  yielding post-filtered energy of the  $j^{\text{th}}$  filter channel as:

$$e_j(x,y) = m_j(x,y) ** g_p(x,y)$$

where,

$$g_p(x,y) = \frac{1}{2\pi\sigma_2^2} e^{-\frac{[x^2+y^2]}{2\sigma_2^2}}$$

The feature vectors computed from the local energy estimates are (i) mean  $\mu[e_j(x,y)]$  and (ii) variance  $\sigma[e_j(x,y)]$ . The mean subtracted magnitude of the four wavelet channels coefficients given in Figure 3 (d), are shown in Figure 4(a)-(d). The corresponding Gaussian post-filtered outputs of the mean subtracted magnitudes are shown in Figure 5(a)-(d). Figure 6(a), (b) shows the energy measure computed from the Gabor filtered coefficients, given in Figure 2 (c), (d). The post-processing stage to compute the features (local mean and variance) from the energy estimates, is the same for both the filtering techniques.

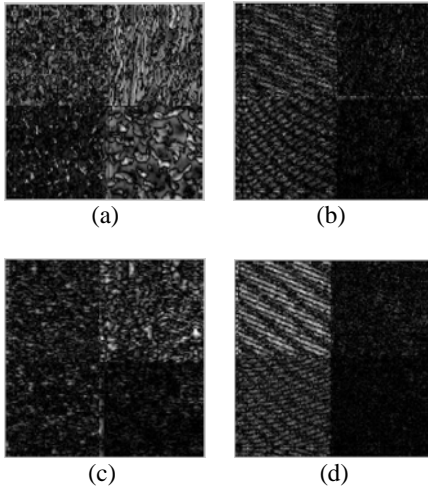


Figure 4: (a)-(d) Mean subtracted magnitude of the DWT coefficients A1, H1, V1, D1 in Figure 3(d)

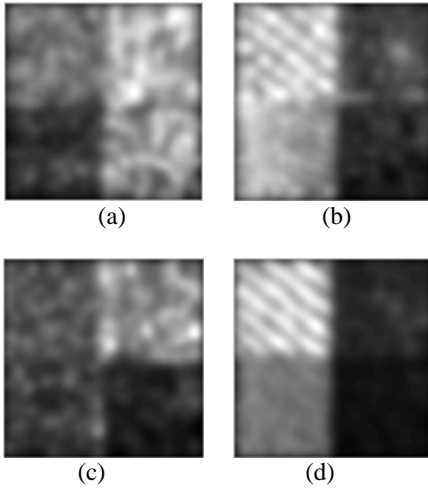


Figure 5: (a)-(d) Energy computed from the post-processed outputs in Figure 4(a)-(d)

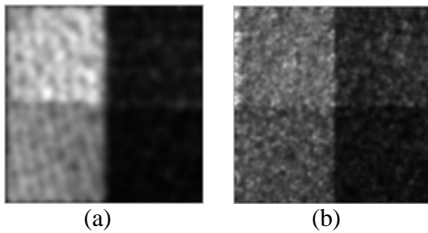


Figure 6: (a), (b) Energy of the Gabor filtered coefficients corresponding to Figure 2(c), (d)

## 4. Classification

There are already a large number of supervised and unsupervised texture segmentation algorithms existing in literature. The difference between supervised and unsupervised segmentation is that supervised segmentation assumes priori knowledge on the type of textures present in the image. We have used here, the fuzzy c-means clustering (FCM) algorithm as an iterative procedure, which is described below:

1. Calculate the fuzzy cluster centers  $v_c^l$  with

$$v_c = \frac{\sum_{m=1}^M (u_{c,m})^w x_m}{\sum_{m=1}^M (u_{c,m})^w}$$

2. Update  $U^{(l)}$  with

$$u_{c,m} = 1 / \sum_{m=1}^C \left( \frac{d_{c,m}}{d_{j,m}} \right)^{\frac{2}{w-1}}$$

where  $(d_{i,m})^2 = \|x_m - v_i\|^2$  and

$\|\bullet\|$  is any inner product induced norm

3. Compare with  $U^{(l+1)}$  in a convenient matrix norm.

If  $\|U^{(l+1)} - U^{(l)}\| \leq \epsilon$  Stop

Otherwise return to Step 1.

where,  $M$  is the size of input data  $\{x_m; m= 1,..M\}$ ,  $C$  is the number of clusters,  $w$  is the fuzzy weighting exponent ( $1 < w < \infty$ ) and  $U^{(l)}$  is the membership function matrix at iteration  $l$ . The value of the weighting exponent  $w$  determines the fuzziness of the clustering decision. A smaller value of  $w$ , i.e.  $w$  close to unity, will lead to a zero/one hard decision membership function, while a larger  $w$  corresponds to a fuzzier output.

While combining the Gabor and wavelet features for providing input to the FCM classifier, we had to ensure that the dimensionality and resolution of the feature vectors were compatible. We have used 8 different Gabor filters and 2 types of wavelet transforms. Daubechies 8-tap and Haar gave 8 features for every pixel, which ensured equal weightage for both the filtering techniques. To ensure resolution compatibility, the wavelet features were upsampled to the same size as that of the Gabor. Results of the FCM classifier using the features of both the filters are described in the next section.

## 5. Experimental results

In this section, we illustrate the performance of the feature extraction methods using several examples of texture images. These images are 8-bit grayscale images of size 256 X 256. The DWT was computed using the two most commonly used filters viz. Daubechies and Haar with level-1 decomposition only. The Gaussian width used for post-filtering was chosen to be 1.5 times the width of the window used for feature extraction. Eight different Gabor filters were selected based on a trial and error method. We observed the energy responses with varying parameter values and the ones that provided contrasting signatures for at least 2-3 different regions were chosen. This set of eight filters for the Gabor filter bank is not optimal, but we avoided specific tuning which could lead to a supervised approach. The parameters of the eight Gabor filters are given in Table I. The width of the Gaussian filter in this case was twice that used for computing the local mean. To ensure unbiased comparison, the segmentation algorithm was kept the same. The feature vector for Gabor filter based classification had a set of 16 features, while the wavelet based technique had 8 each.

Table II shows the results of texture classification on 6 images, each containing four different texture regions. Column (a) in Table II shows the input texture images. Corresponding outputs of the FCM classifier are shown in the columns of Table I with different features obtained using: (b) Gabor, (c) Daubechies, (d) Haar and (e) combined features of Gabor, Haar and Daubechies filters.

The error in segmentation of the FCM classifier is used as a measure to compare the performances of the filters. Error in each region is calculated as:

$$\frac{\text{No of pixels incorrectly classified in that region}}{\text{Total number of pixels in that region}}$$

which is averaged over the entire image. This measure is obtained for all the results shown in Table II and is tabulated for comparison purpose in Table III. Since the size of wavelet coefficients is half of the original image, they were up-sampled to ensure that all features have identical resolution. It was observed that the errors were the least when features from Gabor and wavelet filters were combined together for classification. Figure 7 shows the results of segmentation with an image containing five texture regions, similar to that in Table II.

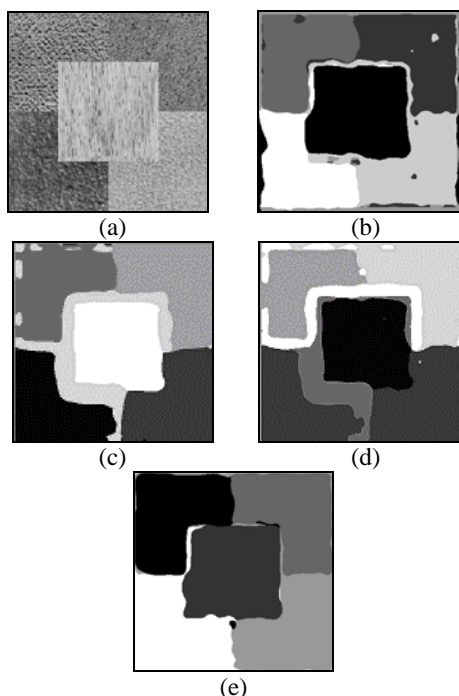


Fig. 7: (a) Input image with five textures. Segmented image with features obtained using (b) Gabor filter only (c) Daubechies filter only (d) Haar transform only and (e) Combined features from Gabor, Daubechies and Haar filters

## 6. Conclusions

The results of our proposed method reveal that a combination of features from two different types of multi-resolution and multi-channel filters (instead of a ‘war’

between Gabor and wavelet) provides superior classification of texture images. The method combines the advantages (or feature discriminability) of both these filters to provide an improved performance. This has been the main objective and aim of this paper. One may down-sample the feature vectors of the Gabor filtered output and obtain feature vectors at lower resolution. However, this obviously results in poorer performance. One can also follow [2], [9] to determine an optimal set of parameters for the Gabor filter to obtain better results.

Filter(i)	1	2	3	4	5	6	7	8
$\sigma_{li}$	2	2	1	2	2	3	1	2
$u_i$	5	1	5	3	2.7	2	4	6
$v_i$	5	4	5	1	2.7	-1	4	-3

Table I: Gabor filter parameters ( $\sigma_{li}$ ,  $u_i$ ,  $v_i$ ) of the 8 filters in the bank

## 7. References

- [1] D. Charalampidis and T. Kasparis. Wavelet-based Rotational Invariant Roughness Features for Texture Classification and Segmentation. IEEE Transactions on Image Processing, Vol. 11, No. 8, pp. 825-837, August 2002.
- [2] D. Dunn, W. E. Higgins, J. Wakeley. Texture Segmentation Using 2-D Gabor Elementary Functions. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, No. 2, pp. 130-149, 1994.
- [3] S. E. Grigorescu, N. Petkov and P. Kruizinga. Comparison of texture Features based on Gabor Filters. IEEE Transactions on Image processing, Vol. 11, No. 10, pp. 1160-1167, October 2002.
- [4] C. Lu, P. Chung and C. Chen. Unsupervised texture segmentation via wavelet transform. Pattern Recognition, Vol. 30, pp. 729-742, 1997.
- [5] Y. Mallet, D. Coomans, J. Kautsky and O. De vel. Classification using Adaptive Wavelets for Feature Extraction. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 10, pp. 1058-1066, October 1997.
- [6] Olaf Pichler, Andreas Teuner and Bedrich J. Hosticka. A comparison of Texture Feature Extraction using adaptive Gabor filtering, pyramidal and tree structured wavelet transforms. Pattern Recognition, Vol 29, No. 5, pp. 733-742, 1996.
- [7] J. Randen and J. H. Husoy, Filtering for Texture Classification: A Comparative Study. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 21, No. 4, pp. 291-310, April 1999.
- [8] E. Salari and Z. Ling. Texture Segmentation using hierarchical wavelet decomposition. Pattern Recognition, Vol. 28, pp. 1819-1824, 1995.
- [9] Thomas P. Weldon, William E. Higgins and Dennis F. Dunn. Efficient Gabor filter design for texture segmentation. Pattern Recognition, Vol. 29, No. 12, pp. 2005-2015, 1996.
- [10] Yegnanarayana B, G. Pavan Kumar and Sukhendu Das. One-Dimensional Gabor Filtering for Texture Edge Detection. Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP '98), 21-23 December 1998, New Delhi, INDIA, pp. 231-237.

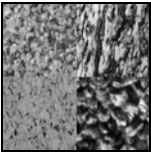




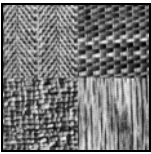
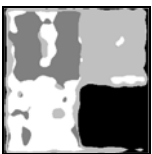



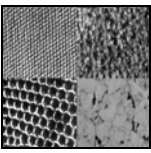




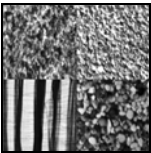




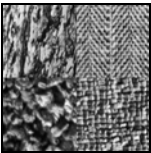
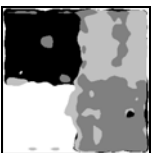



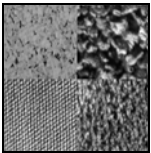




(a) Input Image	Results of classification			
	(b) Gabor	(c) Daubeschies	(d) Haar	(e) Combined features
1 				
2 				
3 				
4 				
5 				
6 				

Table II: Experimental results of classification: (a) Input images. Segmented image with features obtained using (b) Gabor filter only (c) Daubechies filter only (d) Haar transform only and (e) Combined features from Gabor, Daubechies and Haar filters

Image	Error in classification			
	Gabor	Daubechies	Haar	Combined Features
1	5.3650	3.5034	3.7109	2.9541
2	21.6141	37.4853	18.4156	15.6036
3	12.2330	19.9130	18.3625	5.7782
4	27.2964	22.8796	12.0404	10.6079
5	17.5079	21.8721	22.0379	15.1474
6	9.4177	8.3638	15.5295	4.7287

Table III: Classification errors corresponding to the experimental results in Table I.