GLOBAL JOURNAL OF PURE AND APPLIED SCIENCES VOL. 12, NO. 3, 2006: 409 - 414
COPYRIGHT (C) BACHUDO SCIENCE CO. LTD. PRINTED IN NIGERIA. ISSN 1118 - 0579

MULTIRESOLUTION TEXTURE FEATURE EXTRACTION BASED ON PYRAMID WAVELET DECOMPOSITION

G. LOUM

(Received 18 January, 2005; Revision Accepted 13 January, 2005)

ABSTRACT

The goal of texture feature extraction is to obtain a set of texture measures, which can be used to discriminate among different textural pattern classes. In this paper, a new approach to the characterisation of texture properties from standard pyramidal wavelet decomposition is described. Because of the multiresolution data representation of the outputs of wavelet filter bank, we propose to estimate texture features over a pyramidal structure which is composed of the whole levels of the wavelet decomposition. This multiresolution approach of texture feature extraction provides effective texture features for texture classification and is computationally attractive in a progressive segmentation process. Numerical experiments are given to demonstrate the reliability of our new method.

KEYWORDS: Pyramidal structure, wavelet transform, texture classification, texture segmentation.

INTRODUCTION

Wavelet transform has received significant attention recently due to its suitability for a number important signal and image processing tasks. Mallat (*MALLAT, 1989*) first suggested the use of wavelet transform for texture analysis because its discrete version provides a good multiresolution representation of the signal and gives orientation sensitive information for texture characterisation.

During the past decades, many researchers have reported the success of applying discrete wavelet to texture analysis (*ZHANG and TAN, 2002*). Their common approach is equivalent to local linear transform method (*LAWS, 1980 and UNSER, 1986*). Texture image is filtered with a bank of wavelet filters and local texture properties are then characterised by a set of energy measures computed at the output of each filter bank (*CHANG, 1993; SALARI, 1995; UNSER, 1995 and BASHAR, 2003*). Generally these energy values are estimated over a rectangular moving window centered at a given pixel. The size L×L of this window can be fixed (*CHANG, 1993*) or variable according to the level of resolution (*SALARI, 1995*). This approach to texture feature extraction can be viewed as an extension of single resolution technique to multiresolution analysis.

In this article, we propose to compute texture energy values over a pyramidal structure, which is composed of the whole J-levels of wavelet decomposition. This multiresolution approach to texture characterisation seems to be well adapted to the intrinsic multiresolution data representation of the traditional pyramid-type wavelet transform.

This article is organised as follows: In section II we expose briefly the standard pyramid-structured wavelet transform and we define in section III the new structure on which texture feature are extracted. Section IV presents an application of our approach to texture classification and compares its performance with that of local linear transform using Discrete Cosinus Transform (DCT). Finally a progressive texture segmentation algorithm and illustrative examples are proposed in Section V.

STANDARD WAVELET TRANSFORM

Wavelet transform provides a precise and unified framework for multiresolution analysis. The traditional pyramid-type wavelet transform decomposes a signal with a family of orthonormal bases obtained through translation (variable n) and dilatation (variable m) of a kernel function $\psi(x)$ known as the mother wavelet:

$$\Psi_{m,n}(x) = 2^{-m/2}\Psi(2^{-m}x - n)$$

To construct wavelet ψ, we first determine a scaling function $\phi(x)$, which satisfies:

$$\Phi(x) = \sqrt{2}\sum_k h_k \Phi(2x-k)$$

Then, function $\psi(x)$ is determined as:

$$\Psi(x) = \sqrt{2}\sum_k g_k \Phi(2x-k)$$

The coefficients h(k) and g(k) must satisfy the following relation (*DAUBECHIES, 1988*):

$$g(k) = (-1)^k h(1-k)$$

It is convenient to view coefficients h(k) and g(k) as impulse responses of a pair of filters H and G corresponding to a lowpass and highpass filters. H and G are called quadrature mirror in the signal processing literature.
In conventional wavelet decomposition, the pair of filters H and G is applied in both the horizontal and vertical directions, followed by a subsampling by two of each output image. This process provides at the first level of resolution four subimages of the original image I_0(figure 1). I_1 is a coarse or approximate image of I_0 and D^H_1

G. Loum, INP-HB, Departement Genie Electrique et Electronique, BP 1093 Yamoussoukro, Cote d'Ivoire

I_0

$\hookrightarrow I_1$ and D^H_1 D^V_1 D^D_1

$\quad\hookrightarrow I_2$ and D^H_2 D^V_2 D^D_2

$\qquad\downarrow$

Figure 1: Wavelet decomposition process

D^V_1, D^D_1 are the orientation selective detail images respectively in horizontal, vertical and diagonal directions (*MALLAT, 1989*). Due to the subsampling by two at each stage, the spatial resolution of the four subimages decreases accordingly. Decomposition can be iterated from the approximate subimage I_1 to generate the next level of resolution (figure 1). At the end of the process, the stack of subimages of the same type forms a pyramid.

Our goal is to use Wavelet filter bank to match the local texture structures in order to give best discrimination among textured regions with accurate spatial localisation. The success of this approach obviously depends on the judicious choice of the filter bank. Constraints on filter design include perfect reconstruction, finite-length, and the regularity requirement that the iterated lowpass filters involved converge to continuous function. In a previous work, Unser (1995) has pointed out that increasing the regularity of filter bank does not seem to have any real advantage for texture analysis and discrimination. On the other hand, the localisation properties of the analysis filter bank seem to be more important.

In our application, we chose to use the filter bank (H = {1/2; 1/2} and G = {1/2; -1/2}) associated to Haar's wavelet. This choice was made for the following reasons: First, the filters are symmetrical. That means that there is no phase distortion and that the spatial localisation of the wavelet coefficient is well preserved. Second, the use of shorter filters provides faster computation and a good space localisation of the analysis filter bank. Its implementation is equivalent to the local linear transform method using the 2×2 Hadamard transform. Finally in the next section, we will see that Haar's wavelet is very consistent with our approach to texture characterisation.

TEXTURE FEATURE EXTRACTION

The classical wavelet-based feature extraction is related to local linear transform method. In this approach, the local texture properties are characterised by statistics associated with the outputs of wavelet filter bank. For each subimage texture features are estimated over a moving rectangular window with fix or variable size according to the considered level of resolution. This approach is more an extension of single resolution technique to multiresolution analysis than a proper multiresolution technique for texture feature extraction.

In this paper, we propose a new method of texture feature extraction, which takes into account the multiresolution data structure of wavelet decomposition by associating together the different levels of resolution. In that way, texture features will be computed over a pyramidal structure instead of a rectangular window. We think that due to the intrinsic pyramid-type data representation of wavelet transform, this structure is more appropriate for texture feature extraction.

Let us consider the J-level conventional wavelet decomposition of a $2^J \times 2^J$ pixels of an original image. The top level J of the decomposition has a single pixel k. We define the pyramidal structure $P^X_J(k)$ associated to the pixel k, as the set of subimages S^X_j $(1 \le j \le J)$ of a given type X of subimage. Pyramidal structure $P^X_J(k)$ can also be viewed as a collection of rectangular windows S^X_j of sizes $2^{J-j} \times 2^{J-j}$ $(1 \le j \le J)$ varying with the considered level of resolution j. Figure 2 gives an example of a pyramidal

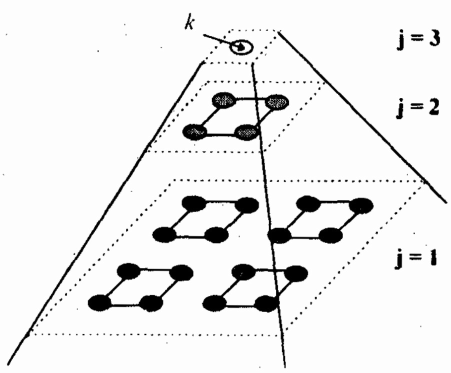


Figure 2: Pyramidal structure of a 3-level wavelet decomposition

structure $P^X_3(k)$ obtained from a 3-level wavelet decomposition of an original image of size $2^3 \times 2^3$

The size of the pyramidal structure $P^X_J(k)$ increases with the top level J. In total, a J-level pyramidal structure has $N_J = \frac{1}{3}(4^J - 1)$ pixels. When J varies, the analysis of spatial interactions on the original image is carried out either over small or large neighbourhoods of $2^J \times 2^J$ pixels. Thus it appears that for any level J, pyramidal structure and Haar's bank filter have exactly the same support of size $2^J \times 2^J$ on the original image. This property makes Haar's wavelet suitable for our texture feature extraction approach. Indeed, each block of $2^J \times 2^J$ pixels can be characterised without any overlapping with another block. This is particularly useful in texture segmentation context where the recovery of different blocks increases classification errors of texture region borders.

Four different pyramidal structures can be obtained from wavelet decomposition. But in our application we use only the three pyramidal structures, which contain orientation sensitive information because it is well known that information about texture pattern orientation is essential to characterise textures.

Many texture features as local texture energy can be computed over the pyramidal structure P_J^X For a given orientation X and for a pixel k at the level J, local texture energy is defined as:

$$E_J^X(k) = \frac{1}{N_J} \sum_{j=1}^{J} \sum_{b \in S_j^X(k)} d^2(b)$$

where j is the current level, S_j^X is the window corresponding to the jth level of the pyramidal structure, $d(b)$ is the value of the pixel b, and N_J denotes the total number of pixels in the pyramidal structure.

At any level J, three local texture energies corresponding to the three orientations (H, V, D) can be computed to form texture feature vector. Note also that local texture energy can be recursively calculated from a level to the next superior.

TEXTURE CLASSIFICATION

Classification is a process where given a textured image, it is assigned to one of a finite number of classes to which the sample belongs. We performed classification experiments using 10 textures displayed in figure 3. The size of the images is 256×256 with 256 levels of grey. We constructed the four-level wavelet decomposition using Haar's filter bank. For each pixel k at level 4, we computed over a 4-level pyramidal structure of N_4=85 pixels, 3 local texture energies (E^V_4, E^H_4 and E^D_4) as components of texture feature vector.

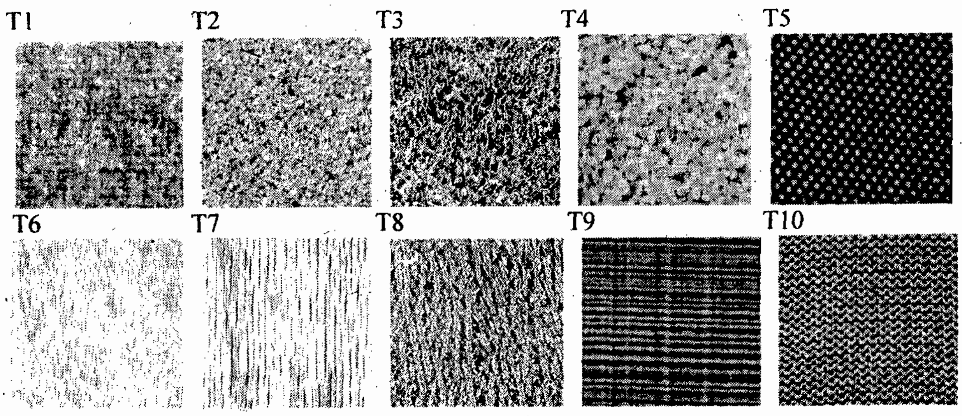


Figure 3: Textures of our experiment

For each texture a total of 256 independent feature vectors was evaluated. The discrimination function used is Euclidean distance. We choose to compare the performance of our method (M1) and those of local linear transform using Discrete Cosinus Transform (DCT). To make feature set comparable, we implement the DCT using its vertical (V), horizontal (H) and diagonal (D) edge detectors, which have the same orientations than those of wavelet transform. Local texture energy is then computed at each filter output over a 15×15 moving rectangular window (method M2).

$$V = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad D = \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

The classification results for the two methods are given in Table 1. We can note that our method is particularly sensitive to structural textures, which have high concentration of localised spatial frequencies. In contrast, DCT features computed over a rectangular window yield better performance principally for the class of so-called micro-textures (T1-T3). Globally, our texture feature extraction scheme outperforms the DCT one. This result confirms that multiresolution method is more efficient than a single-level analysis, which focus on relatively small neighbourhoods. In the following section we discussed the use of the pyramidal structure in a progressive texture segmentation process.

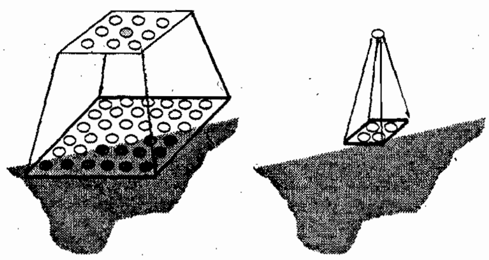
Table 1: Percent of correct classification of ten textures

Textures	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Total Score (%)
M1 (%)	92.1	87.1	87.5	92.2	98.4	93.8	91.8	90.6	91.4	98.4	92.33
M2 (%)	94.5	90.9	91.0	91.0	86.7	94.1	84.4	83.6	93.8	92.3	90.23

TEXTURE SEGMENTATION

Texture segmentation consists in partitioning a given image into connected regions of homogenous texture. The most difficult problem is to correctly determine the boundaries between textures because generally, features extracted in the boundary regions are representative of two or several textures.

In a multiresolution context, the definition of an n×n rectangular window at level J, corresponds to a block of size $2^J n \times 2^J n$ on the original image. Such block becomes very large when J and n increase (figure 4.a) and therefore can not help to accurately describe boundaries between texture. To reduce segmentation errors, we propose to estimate texture features over a pyramidal structure, which corresponds to a less small block of size $2^J \times 2^J$ on the original image (figure 4.b). Moreover the size of the pyramidal structure varies automatically with the level of resolution (figure 5). This adaptable size is very useful for a progressive segmentation process wnich moves according to "coarse to fine" resolution strategy.



(a) (b)

Figure 4: Comparison of the supports of a rectangular window (a) and pyramidal structure (b)

The segmentation process begins by the construction of a J-level Haar's wavelet decomposition of the original image of size $2^N \times 2^N$. Then for each pixel k at the top-level j=J, the 3 local texture energies are evaluated over a pyramidal structure to form texture feature vector associated to the pixel k. We assume that the number of cluster c is known. At level j=J, texture feature vectors are used in a fuzzy C-means clustering algorithm to segment the textured image into c labelled regions. We obtain a coarse segmentation of the original image. Then a relaxation process is conducted on the segmented image in order to integrate a kind of spatial relationship, which is not taken into account in the

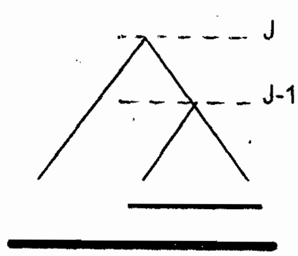
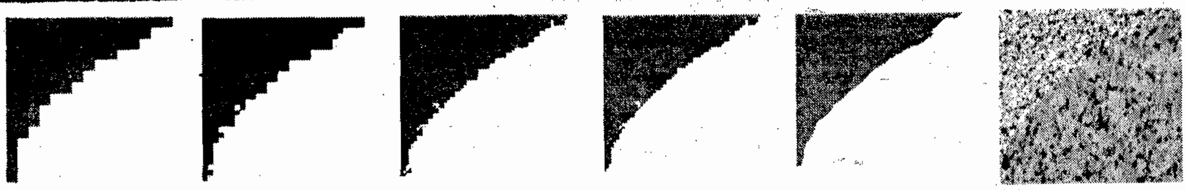


Figure 5: Variable size of the pyramidal structure



j=4 j=3 j=2 j=1 final result (j=0) original image

Figure 6: Progressive texture segmentation results with 2 textures of sand

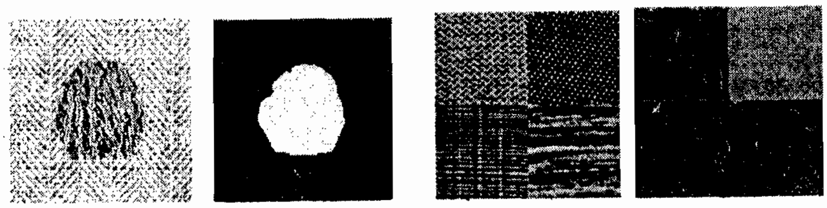


Figure 7: Texture segmentation result with 2 textures and 4 textures

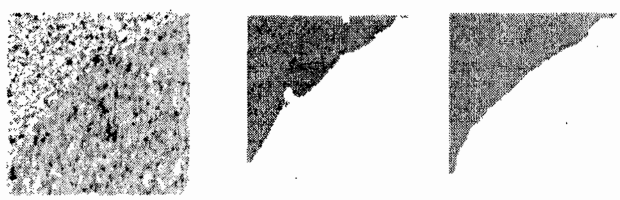
Table 2: Comparison of segmentation process using "pyramidal" and 3×3 "rectangular" windows

	Pyramidal structure	**3×3 rectangular window**
Correct classification rate (%)	(fine sand) 97.3 (coarse sand) 97.8	(fine sand) 93.5 (coarse sand) 94.8
Computation time (s)	33.5	47

clustering phase. Thus a vote is held among the eight connected neighbours to detect the ambiguous pixels, (e.g. pixels which are isolated or which belong to the region borders).

To improve the first result, the segmented image is expanded from size $2^{N-j} \times 2^{N-j}$ to size $2^{N-j+1} \times 2^{N-j+1}$ corresponding to the level j=j-1. Only the clustering of the ambiguous pixels will be reported at this new current level j. In this way, successive results of segmentation are propagated from the top to the bottom of the wavelet decomposition structure. At the end of the process, a 3×3 median filter is used to smooth the final result.

Figure 6 illustrates the different phases of segmentation process with an image composed of coarse and fine sands. The segmentation process begins at level j=J=4 and is iterated until level j=1. We can see that the first result (j=4) is gradually improved. Finally a 3×3 median filter is applied to obtain the segmentation result of the original image. Figure 7 gives also the results of the segmentation of two different texture images.



Original image (a) (b)

Figure 8: Segmentation results with 3×3 rectangular window (a) and pyramidal structure (b)

We have compared our results to those of the classical method using a moving 3×3 rectangular window at different levels. Segmentation results are given in Table 2 and in Figure 8. Our method, which uses a pyramidal structure, gives better clustering results as well as good computation time. As we expect, we can note that when rectangular window is used, classification errors are principally located in the texture region borders.

Table 3: Influence of the top level J

	Textures	J=3	J=4	J=5
Correct classification rate(%)	(fine sand)	92,6	97.3	93 2
	(coarse sand)	93,8	97.8	95.2
Computation time (s)		37	33.5	35

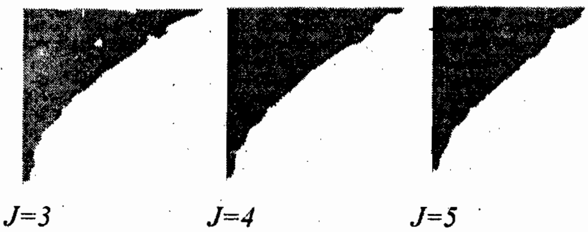


Figure 9: Influence of the top level of wavelet decomposition

We have also studied the influence of the top level J on the segmentation results (Table 3). We can note that when J value is relatively small, the clustering gives not satisfactory results (figure 8). It confirms that texture features calculated over a window with small size are not robust. In other hand, increase excessively J value does not guaranty best performance. The choice of J=4 seems to be relevant in many applications as reported by others studies (*CHANG, 1993; SALARI, 1995*).

CONCLUSION

In this paper, a multiresolution wavelet-based feature extraction has been proposed. After wavelet decomposition, texture features was computed over a pyramidal structure. These multiresolution features are relevant to discriminate between different textures. Experimental results of a texture classification algorithm have been presented which demonstrate the performance of the feature extraction method. Segmentation experiments were also conducted using the advantage of the pyramidal data representation As the size of the support of the pyramid structure varies according the resolution, a progressive texture segmentation algorithm has been proposed. The advantage of this process is to provide adequate segmentation at much lower cost than a single resolution method. Note that the filter bank associated to Haar's wavelet is very consistent with our approach because it leads to a segmentation process without non-overlapping regions.

REFERENCES

Bashar, M. K., Matsumoto, T., and Ohnishi, N., 2003. Wavelet transform-based locally orderless images for texture segmentation. Pattern Recognition Letters, 4(15): 2633-2650

Chang, T. and Kuo, C. C. J., 1993. Texture Analysis and Classification with Tree-Structured Wavelet Transform. IEEE Transactions on Image Processing, 2(4): 429-441.

Daubechies, I., 1988. Orthonormal bases of compactly supported wavelets. Com. of Pure and Applied Math. 41: 909-996.

Laws, K.I., 1980. Textured image segmentation. Ph.D Thesis, Rept. 940, Image Processing Institute, University of southern California.

Mallat, S.G., 1989. A Theory of Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 11: 674-693.

Salari, E. and Ling, Z., 1995. Texture Segmentation Using Hierarchical Wavelet Decomposition. Pattern Recognition, 28(12): 1819-1824.

Unser, M., 1986. Local linear transforms for texture measurements. Signal processing. 11: 61-79.

Unser M., 1995. Texture Classification and Segmentation Using Wavelet Frames. IEEE Transactions on Image Processing. 4(11): 1549-1560.

Zhang J. and Tan T., 2002. Brief review of invariant texture analysis methods. Pattern Recognition. 35(3): 735-747.