ISSN:0975-9646

Mansi Saraswat et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (6) , 2013, 775-782

Object Recognition Using Texture Based Analysis

Mansi Saraswat[#], Anil Kumar Goswami[*], Aastha Tiwari[#]

[#]*Banasthali University, India*
[*]*DRDO, Delhi, India*

***Abstract*-Object recognition plays an important role in the area of image processing and target based applications. In order to identify an object we must have its features in the form of a feature vector. This can be achieved by feature extraction. There are various ways of extracting features of an image. It can be based on color, texture or shape. The aim of this paper is to study and compare the different texture based approaches for object recognition and feature extraction. GLCM and Haar wavelet transform are the most primitive methods for texture analysis. In this paper two more techniques based on their fusion have been considered. These techniques have been tested on sample images and their detailed experimental results along with the method of implementation have been discussed.**

***Keywords*- GLCM, Haar, fusion, texture, object recognition**

I. INTRODUCTION

Texture is that inborn property of all surfaces that describes visual patterns and contains important information about its arrangement. In short, texture describes the distinctive physical composition of a surface [1]. Since an image is made up of pixels, texture can be defined as an entity consisting of mutually related pixels and group of pixels. This group of pixels is called as texture primitives or texture elements (texels) [2]. Hence, texture based techniques can be applied only on a group of pixel and not a single pixel. There is no visible inter-object part-wise correspondence for texture based objects. These objects are better described by their texture than the geometric structure of reliably detectable part. Buildings, roads, trees and skies are texture based objects. Sometimes when we are using color based segmentation shadow is detected as a different object. But if we use texture based object detection technique shadow can be completely eliminated [3].

Texture is a prominent and crucial feature for content-based image retrieval applications, image shape identification and image segmentation through pattern recognition and similarity matching. Periodicity, scalability, coarseness, inherent direction and pattern complexity are considered as the most perceptually distinct properties of texture [4]. Texture depends on the distribution of intensity over the image rather than being defined for a separate pixel. A lot of work has been done on analysis, classification and segmentation of texture but still it is considered as an active research topic with numerous algorithms being developed based on different techniques.

Section 2 of the paper provides the literature survey in this area. In the section 3, number of texture analysis techniques will be explained and discussed. Section 4 gives the methodology. Section 5 provides overview of experiments performed and results obtained using these techniques. Section 6 provides the conclusion.

II. LITERATURE SURVEY

The texture is very important cue in region based segmentation of images. Texture features play a very important role in computer vision and pattern recognition [5]. Apart from that, texture has been widely involved in many real life applications such as remote sensing (Ruiz et al., 2004), biomedical image processing (Tuceryan et al., 1998), content based image retrieval (Manjunath and Ma, 1996), and rock and wood species classification (Tou, 2009) [6]. It is essential to find the set of texture features with good discriminating power to design an effective algorithm for texture classification.

Many texture based feature extraction methods are possible. Some of the most popular methods are gray level co-occurrence matrix, Gabor filters, Haar filters, Daubechies filters etc [3]. A fusion of two or more above mentioned techniques can also be used to get better results.

Although there are variants of texture analysis methods, among them, gray level co-occurrence matrix (GLCM) algorithm 1986 is probably the most commonly adopted for remote sensing. GLCM is very suitable for finding texture information in images of natural scenes and performs well in classification [7]. Many features have been proposed to precisely define the natural texture properties. Tamura proposed six features. Those features are coarseness, contrast, directionality, line-likeness, regularity and roughness. Gabor features and wavelet features are widely used in image retrieval system and gives good result [3]. There can be different wavelet based techniques. An image that undergoes Haar wavelet transform will be divided into four bands at each of the transform level and its mean and standard deviation are calculated for each band.

The fusion of GLCM and Haar wavelet decomposition can be used to form a single technique. This is done by replacing the low pass filter and the high pass filter with the four bands of the Haar wavelet transform.

III. TEXTURE ANALYSIS TECHNIQUES

Texture features can be found using methods as GLCM, Haar Wavelet Decomposition and Wavelet GLCM fusion etc.

A. Gray Level Co-Occurrence Matrix (GLCM)

The identification of specific textures in an image is achieved primarily by modeling texture as a two-dimensional gray level variation. This two dimensional array is called as Gray Level Co-occurrence Matrix (GLCM) [8]. The GLCM (Gray-level co-occurrence matrix) is a statistical image analysis technique that is used to estimate image properties related to second-order statistics. It defines both the structural as well as spatial properties of an image texture. It considers the relation between two neighboring pixels in one offset, where the first pixel is called reference and the second one the neighbor pixel. When the image is transformed into the co-occurrence matrix, the neighboring pixel can be taken in any of the eight defined directions. Generally, 0^0, 45^0, 90^0 and 135^0 are used. However their reverse directions (negative direction) can also be taken into account. Therefore general GLCM depends upon the directionality [7]. The probability of the pair of pixels having gray levels i and j, with an inter sample distance d and direction θ can be represented by the GLCM element G (i, j, d, Θ).

Various features can be extracted from the GLCM. The mean describes the average gray level for each region while standard deviation is more informative [9]. Homogeneity is measure for uniformity of co-occurrence matrix, dissimilarity measures how different the elements of the co-occurrence matrix are from each other, entropy measures randomness, whereas energy and ASM measure the extent of pixel pair repetitions and pixel orderliness [7].Contrast is the measure of vividness of the texture pattern [1] and correlation gives the linear dependency of gray levels on those of neighboring pixels.

There are several difficulties in the GLCM technique. There is no pre defined method for selection of the displacement vector and computing co-occurrence matrices for different values is tedious. For a given image, a large number of features can be computed from GLCM. So, a feature selection method must be used to select the most relevant features.

B. Haar Wavelet

Haar wavelet transforms the image from the space domain to a local frequency domain. It consists of a series of low pass and high pass filters known as the filter banks. The discrete wavelet transform (DWT) uses filter banks to perform the wavelet analysis and divides the image into various frequency bands. The Haar wavelet technique decomposes the image into 4 sub-bands, at each level, labeled as LL, HL, LH and HH. The LL band gives the approximate image and can be decomposed further. The other 3 bands are the detail bands and contain information about the directions.

The wavelet transform transforms the image into a multi-scale representation with both spatial and frequency characteristics. This allows for effective multi-scale image analysis with lower computational cost [1]. It provides orientation sensitive information and there is no loss of information in the process of decomposition.

One disadvantage of Haar wavelets is that it tends to produce large number of signatures for all windows in image [10]. The other disadvantage of the Haar wavelet is that it is not continuous, and therefore not differentiable. This property can, however, be an advantage for the analysis of signals with sudden transitions, such as monitoring of tool failure in machines [11].

The figure below shows the decomposition of image matrix at each level.

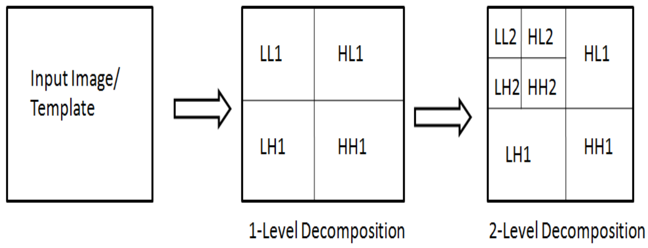

Fig. 1 Two-level haar wavelet decomposition of image

Let us consider two 2×2 matrices: H and Y.

If

$$H = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Then,

$$Y = \begin{pmatrix} a+b+c+d & a-b+c-d \\ a+b-c-d & a-b-c+d \end{pmatrix}$$

where Y represents the Haar Transform of H.

C. Wavelet GLCM Fusion

The above two techniques of GLCM and wavelet can be merged together to describe the texture features of an image because in Haar wavelet, the bands are strongly correlated with the orientation elements in GLCM calculation. Furthermore, when GLCM is used through averaging and differencing filters it is insufficient because a temporary array is needed to store the results of the bands, so Haar wavelet transform is used and it can be done directly on the input image. Its aim is to reduce the computational burden of the original GLCM [12]. It has been shown that GLCM based Haar computation performs slightly better than the original GLCM. It results in improved classification accuracy due to smoothening process [12]. It is robust to rotational variations and demonstrates an improvement in recognition accuracy.

IV. METHODOLOGY

The system is based on selecting an image as an input image. A particular area from this image is selected as the query template. A feature vector is obtained for this template by using any of the above mentioned techniques. A window of fixed size moves through the entire image and calculates feature vector for each of the template which is compared with the query feature vector and the top 50 results or results based on the threshold are displayed. The block diagram of the system is as shown in the figure below:

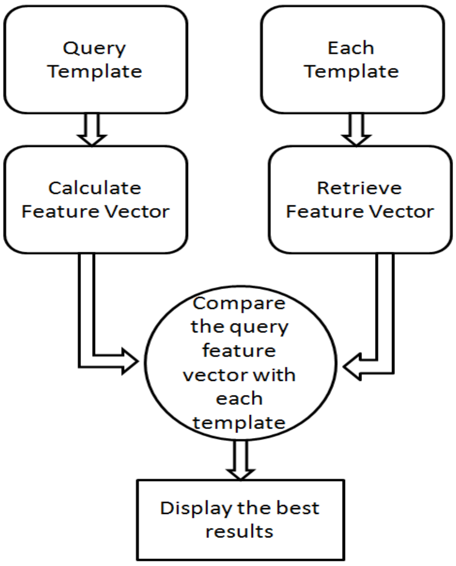

Fig. 2 Block diagram of the system

The implementation details of various techniques are as follows:

A. GLCM

In GLCM technique firstly a multispectral image is converted to gray image and then the corresponding GLCM is obtained. To find the GLCM, the first step is to define the displacement vector dx and dy and the directionθ. For a position operator P, we can define a matrix P (i,j) that counts the number of times a pixel with gray-level i occurs at position P from a pixel with grey-level j. The Gray-Level Co-occurrence Matrix (GLCM) C can be obtained by normalizing the matrix P.

The block diagram showing the implementation of GLCM is as below:

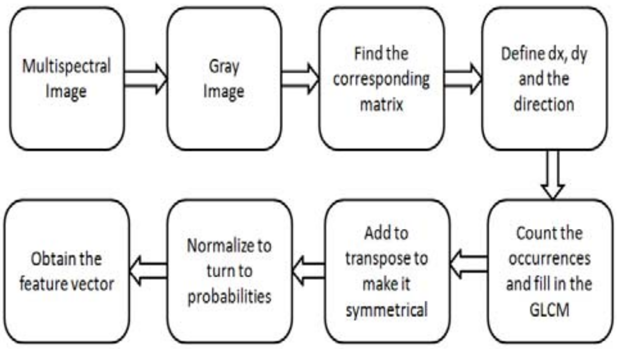

Fig. 3 Block diagram of GLCM technique

The feature vector obtained consists of the following features [13]:

TABLE I
GLCM FEATURES

S.NO	FEATURE	FORMULA
1	MeanX	$\mu_i = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} i\, p(i,j)$
2	MeanY	$\mu_j = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} j\, p(i,j)$
3	Standard DeviationX	$\sigma_i = \sqrt{\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)(i-\mu_i)}$
4	Standard DeviationY	$\sigma_j = \sqrt{\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)(j-\mu_j)}$
5	Contrast	$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)(i-j)^2$
6	Dissimilarity	$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)\lvert i-j\rvert$
7	Homogeneity	$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \frac{p(i,j)}{1+(i-j)^2}$
8	Entropy	$-\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)\log p(i,j)$
9	Energy	$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} p(i,j)^2$
10	Correlation	$\sum_{i=0}^{n-1}\sum_{j=0}^{n-1} \frac{\lvert ijp(i,j)-\mu_i\mu_j\rvert}{\sigma_i\sigma_j}$

The algorithm for the techniques is:

1. Inputs –
   a. ImageTemplateHeight
   b. ImageTemplateWidth
   c. NoOfBands
   d. ImageTemplateDataMatrix
      [Height] [Width][NoOfBands]
2. Convert the image template to gray Image
      For i->0 to Height-1
      for j->0 to Width-1
        GrayImage[i][j]=
      (0.30*ImageTemplateDataMatrix[i][j][0]+
      0.59*ImageTemplateDataMatrix[i][j][1]+
      0.11 * ImageTemplateDataMatrix[i][j][2])
3. Find max and min values from the
      GrayImage[i][j]
4. size=max-min+1

5. dx=1 dy=0 x_red=0 y_red=0 total_r=0
    Define a GLCM probability matrix[size][size]
    For i->0 to Height-1
     for j->0 to Width-1
       x_red = Gray_Image[i][j]
       y_red = Gray_Image[i][j]
       prob_r_mat[x_red-min][y_red-min]++
       total_r++
6. Find the GLCM parameters
   a. For i->0 to size
      for j->0 to size
        if(total_r!=0)
        {
      prob_r_mat[i][j]=prob_r_mat[i][j]/total_r
      EnergyR+=prob_r_mat[i][j]*prob_r_mat[i][j]
      InertiaR+=((i-j)*(i-j))*prob_r_mat[i][j]
        if(prob_r_mat[i][j]!=0.0)
EntropyR-=
prob_r_mat[i][j]*Math.log(prob_r_mat[i][j]
DissimilarityR+=prob_r_mat[i][j]*Math.abs(i-j)
HomogeneityR+=prob_r_mat[i][j]/(1+((i-j)*(i-j)))
MeanXR+=(i)*prob_r_mat[i][j]
MeanYR+=(j)*prob_r_mat[i][j]
   b. For i->0 to size
      for j->0 to size
        VarXR+= prob_r_mat[i][j]*
           (Math.abs(i-MeanXR))
        VarYR+= prob_r_mat[i][j]*
           (Math.abs(j-MeanYR))
   c. StDevXR=Math.sqrt(VarXR)
      StDevYR=Math.sqrt(VarYR)

   d. For i->0 to size
      for j->0 to size
CorrelationR+=Math.abs(((i-MeanXR)*
      (j- MeanYR)*prob_r_mat[i][j])
      / (StDevXR*StDevYR))
7. Assign all the values to GLCMParameters[]
8. Return GLCMparameters

B. Haar Wavelet

Haar wavelet method is based on converting the image into its corresponding gray image. The gray image is divided into 4 bands. Mean and standard deviation for all the 3 bands except the LL band is calculated and stored in the feature vector. LL band is again divided into 4 bands. The above procedure is repeated recursively until the depth of partition is reached. The feature vector in this method consists of the following features:

$$\text{Mean}=\frac{1}{(N*M)}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}P[i][j]$$

Standard deviation

$$=\sqrt[4]{\left(\frac{1}{(N*M)}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\left(P[i][j]-Mean\right)^{2}\right)}$$

The algorithm for implementation is:
1. Inputs –
   a. ImageTemplateHeight
   b. ImageTemplateWidth
   c. NoOfBands
   d. ImageTemplateDataMatrix[Height][Width][NoOfBands]
   e. Depth of Partition
2. Convert the image template to gray Image
      for i->0 to Height-1
      for j->0 to Width-1
        Gray Image[i][j]=
(0.30*ImageTemplateDataMatrix[i][j][0]
+0.59*ImageTemplateDataMatrix[i][j][1]
+0.11*ImageTemplateDataMatrix[i][j][2])
3. Check for the correct depth of partition and assign it maximum possible value if it is more than maximum
4. Calculate Transform
   i.)If Height or width is odd, make it even by subtracting 1.
ii.) Decompose the image into LL,HL,HH and LH band.
iii.)Normalize the value of transformed matrix
iv.)if partitions are less than the depth of partition
   a. calculate mean and standard deviation for the LH,HL and HH band
   b. Store corresponding values in feature_vector
   c. Height=Height/2
   d. Width=Width/2
   e. Calculate Transform for LL band (step 4)
ii.) Decompose the image into LL,HL,HH and LH band.
iii.)Normalize the value of transformed matrix
iv.)if partitions are less than the depth of partition
   a. calculate mean and standard deviation for the LH,HL and HH band
   b. Store corresponding values in feature_vector
   c. Height=Height/2
   d. Width=Width/2
   e. Calculate Transform for LL band (step 4)
v.)else
   calculate mean and standard deviation for all the 4 bands
5. Output feature_vector

C. Wavelet and GLCM Fusion

The above two techniques can be merged together to form a new technique. There can be two approaches to this:

1) Technique 1: The input image is firstly converted to gray image pixel by pixel. Haar wavelet transform is applied on it to get the four frequency bands. GLCM technique is now used on two of the detailed bands LH and HL. LL band is not considered because it is nothing but the approximation of the original image and HH band gives the features which are not much significant. Therefore, it gives us a set of 20 features (10 for each band). This can be represented diagrammatically as:

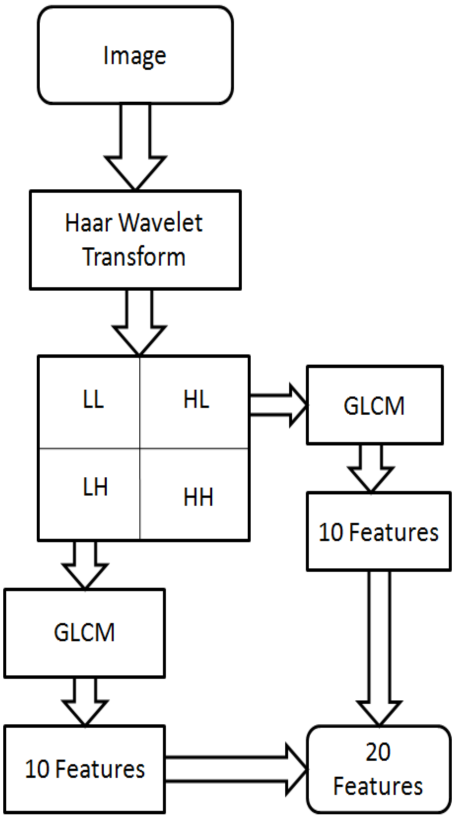

Fig. 4 Block diagram of technique 1

The algorithm for implementation is:

1. Inputs –
   a. ImageTemplateHeight
   b. ImageTemplateWidth
   c. NoOfBands
   d. ImageTemplateDataMatrix [Height] [Width][NoOfBands]
2. For i->0 to Height-2, i+=2
   for j->0 to Width-2, j+=2
   i) WaveletMatrix[i/2][j/2]=0;
   WaveletMatrix[i/2][(Width/2)+(j/2)]=0;
   WaveletMatrix[(Height/2)+(i/2)][j/2]=0;
   WaveletMatrix[(Height/2)+(i/2)] [(Width/2+(j/2)]= 0;
   ii) a= ((0.2989* TemplateMatrix[i][j][0]+ (0.5870*TemplateMatrix[i][j][1])+ (0.1140*TemplateMatrix[i][j][2]))
   b= ((0.2989* TemplateMatrix[i][j+1][0])+ (0.5870*TemplateMatrix[i][j+1][1])+ (0.1140*TemplateMatrix[i][j+1][2]))
   c= ((0.2989* TemplateMatrix[i+1][j][0])+ (0.5870*TemplateMatrix[i+1][j][1])+ (0.1140*TemplateMatrix[i+1][j][2]))
   d= ((0.2989* TemplateMatrix[i+1][j+1][0])+ (0.5870*TemplateMatrix[i+1][j+1][1])+ (0.1140*TemplateMatrix[i+1][j+1][2]))
   iii) WaveletMatrix[i/2][j/2]=(a+b+c+d)/1.414
   WaveletMatrix[i/2][(Width/2)+(j/2)]= (a+b-c-d)/1.414
   WaveletMatrix[(Height/2)+(i/2)][j/2]= (a-b+c-d)/1.414
   WaveletMatrix[(Height/2)+(i/2)][(Width/2)+ (j/2)]= (a-b-c+d)/1.414
3. Find max and min from WaveletMatrix
4. For i->0 to Height/2
   for j->0 to Width/2
   WaveletMatrix[i][j+(Width/2)]= (WaveletMatrix[i][j+(Width/2)]-Min)/ (Max-Min))*255)
   WaveletMatrix[i+(Height/2)][j]= (WaveletMatrix[i+(Height/2)][j]-Min)/ (Max-Min))*255)
   D1[i][j]=WaveletMatrix[i][j+(Width/2)]
   D2[i][j]=WaveletMatrix[i+(Height/2)][j]
5. Find GLCM features of D1 and D2 matrix and assign to WaveletGLCMFeatures[]
6. Return WaveletGLCMFeatures[]

2) Technique 2: Firstly the input image is converted to the gray image. Then α and β are obtained as follows:
If *I (x, y)* is the image then let $I(x,y) = \alpha \; and \; I(x + d\emptyset_0, y - d\emptyset_1) = \beta$.
The table shows α and β values at different orientations based on wavelet GLCM fusion [12].

TABLE II
FORMULAE FOR α and β

Wavelet Band(ϕ)	$\alpha_\emptyset$ & $\beta_\emptyset$
Horizontal (hor)	$\alpha_{hor} = \frac{1}{2}I(x,y) + \frac{1}{2}I(x,y+1)$ $\beta_{hor} = \frac{1}{2}I(x+d,y) + \frac{1}{2}I(x+d,y+1)$
Vertical (ver)	$\alpha_{ver} = \frac{1}{2}I(x,y) + \frac{1}{2}I(x+1,y)$ $\beta_{ver} = \frac{1}{2}I(x,y+d) + \frac{1}{2}I(x+1,y+d)$
Diagonal (dia)	$\alpha_{dia} = \frac{1}{2}I(x,y) + \frac{1}{2}(x+d,y+d)$ $\beta_{dia} = \frac{1}{2}I(x+d,y) + \frac{1}{2}I(x,y+d)$

Then the GLCM features for all the 6 above alpha and beta are calculated. Hence a set of 60 features is obtained.
The algorithm for implementation is:

```
1. Inputs –
     a. ImageTemplateHeight
     b. ImageTemplateWidth
     c. NoOfBands
     d. ImageTemplateDataMatrix
          [Height] [Width][NoOfBands]
2. Convert the image template to gray Image
       For i->0 to Height-1
       for j->0 to Width-1
          GrayImage[i][j]=
       (0.30*ImageTemplateDataMatrix[i][j][0]+
       0.59*ImageTemplateDataMatrix[i][j][1]+
       0.11 * ImageTemplateDataMatrix[i][j][2])

3. if(Height%2!=0 || Width%2!=0)
        ImageTemplateDataMatrix[Height][Width]
        [NoOfBands]=0;
4.   for i->0 to Height-1, i+=2
     for j->0 to Width-1, j+=2
          i. alpha_hor[row][column]=
(int)(Gray_Image[i][j]/2+Gray_Image[i][j+1]/2)
             beta_hor[row][column]=
(int)(Gray_Image[i+1][j]/2+Gray_Image[i+1][j+1]/2)
             alpha_ver[row][column]=
(int)(Gray_Image[i][j]/2+Gray_Image[i+1][j]/2)
             beta_ver[row][column]=
(int)(Gray_Image[i][j+1]/2+Gray_Image[i+1][j+1]/2)
             alpha_diag[row][column]=
(int)(Gray_Image[i][j]/2+Gray_Image[i+1][j+1]/2)
             beta_diag[row][column]=
(int)(Gray_Image[i+1][j]/2+Gray_Image[i][j+1]/2);
          ii. if(counter==Height/2)
                  row++
                  column=0;
5. Find GLCM for each alpha and beta
6. Store in feature vector
7. Output Feature Vector
```

The size of feature vector depends upon the technique selected. The table below shows the size of feature vector obtained by the techniques described in this paper:-

TABLE III
SIZE OF FEATURE VECTOR

S.NO	TECHNIQUE	SIZE OF FEATURE VECTOR
1	GLCM Features	10
2	Haar Wavelet	2*(Depth*3+1)
3	WaveletGLCM Fusion Technique 1	20
4	WaveletGLCM Fusion Technique 2	60(10 Features For Each Of The 6 Bands)

Hence all the above techniques can be shown as:

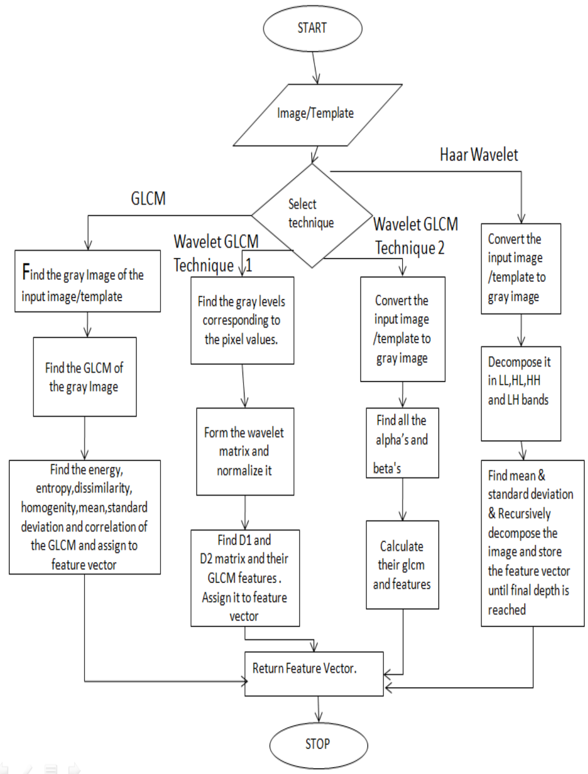

Fig. 5 Flowchart of all the techniques

V. RESULTS AND DISCUSSION

The above techniques were tested on two sample images in Fig. 6 and Fig. 7.

Fig. 6 Test image 1

Fig. 7 Test image 2

The experimental details are provided in the table below:

TABLE IV
TEST DATA

Category	**Test Image 1**	**Test Image 2**
Image Width	500	600
Image Height	500	600
Query Template Width	60	70
Query Template Height	60	70
Pixel Move	10	15
No Of Results	Top 20	Top 20

The system had 64-bit operating system with 4 GB RAM and Intel(R) Xeon(R) E5504 @ 2.00GHz 2.00GHz

The table given below shows the time taken by each technique and the results obtained in each case have been shown separately.

TABLE V
TIME TAKEN by EACH TECHNIQUE

S.No.	Technique	Time Taken (HH:MM:SS:MS)	
		Figure 1	**Figure 2**
1	GLCM	00:00:03:385	00:00:02:556
2	Wavelet	00:00:00:905	00:00:00:717
3	Fusion 1	00:00:00:983	00:00:01:014
4	Fusion 2	00:21:27:826	00:20:00:610

The results obtained when different techniques were applied are as shown below:

GLCM:

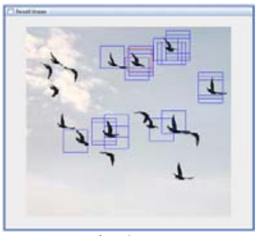

Fig. 8

Fig. 9

Wavelet:

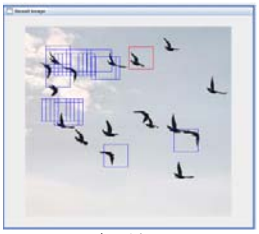

Fig. 10

Fig. 11

Wavelet GLCM fusion 1:

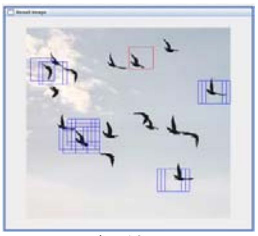

Fig. 12

Fig. 13

Wavelet GLCM fusion 2:

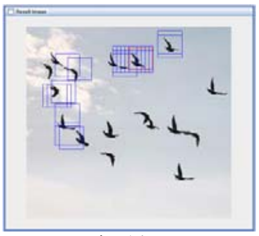

Fig. 14

Fig. 15

The input images in both the test cases are given by Fig. 6 and Fig. 7. The red window highlights the query template whereas blue windows display the results obtained in each case. For Fig. 6, top 20 results have been shown and for Fig. 7, top 30. According to the results, Haar Wavelet takes the minimum time for execution although the results obtained are not very accurate as compared to other techniques. GLCM gives good results and identifies more objects but takes considerable time in execution. The fusion of both the techniques is better as it captures most of the objects and the time complexity is also very less as compared to GLCM. Technique 2 of fusion takes longer to execute but results are more precise than Haar Wavelet.

VI. CONCLUSION

Different texture based approaches have been discussed in this paper. These techniques can be applied only when an image has textural properties. Each technique has its pros and cons. The results depend upon a lot of factors like template size, pixel moves, system configuration, spatial distribution of pixels and the complexity of input image. GLCM gives more specific results as compared to Haar Wavelet but takes more time. Hence, when both the techniques were combined, they gave the best results in least time. It also increases the classification accuracy and reduces the computational burden.

Texture analysis has been employed in a variety of applications such as remote sensing, medical image processing, document processing, and classification of land use categories. The techniques described in this paper can play vital role in these areas.

REFERENCES

[1] Mohamed Gebril, Ruben Buaba, Abdollah Homaifar, Eric Kihn and Mikhail Zhizhin, “Structural Indexing of Satellite Images using Texture Feature Extraction for Retrieval”, *IEEEAC* paper #1458, Version 5, August 2010

[2] G. N. Srinivasan, and Shobha G.,”Statistical Texture Analysis”, *Proceedings Of World Academy Of Science, Engineering And Technology Issn 2070-3740 Volume 36* ,December 2008

[3] Dipankar Hazra,”Texture Recognition with combined GLCM, Wavelet and Rotated Wavelet Features”, *International Journal of Computer and Electrical Engineering, Vol.3, No.1*, February, 2011

[4] Ju Han , Kai-Kuang Ma, “Rotation-invariant and scale-invariant Gabor features for texture image retrieval”, *Image and Vision Computing 25 1474– 1481*, 2007

[5] S. Arivazhagan, L. Ganesan, S.Selvanidhyananthan, ”Image Retreival using Shape Feature”, International Journal Of Imaging Science and Engineering(IJISE), Volume 1, Number 3, July 2007

[6] Alvin Yeo Zong Yun,”Texture Classification Using Wavelet Transform And Support Vector Machines”, *Faculty of Information and Communication Technology (Perak Campus)* ,May 2011

[7] Y N Mamatha and A.G Ananth , “Texture Based Content Retrival For Satellite Images Using Filter Technique”

[8] Dr. H.B.Kekre, Sudeep D. Thepade, Tanuja K. Sarode and Vashali Suryawanshim, “Image Retrieval using Texture Features extracted from GLCM, LBG and KPE”, *International Journal of Computer Theory and Engineering, Vol. 2, No. 5*, October, 2010

[9] U.S.N.Raju, V. Vijaya Kumar, A.Suresh, M.Radhika Mani, “Texture Description using Different Wavelet Transforms based on Statistical Parameters”, Proceedings of the 2nd WSEAS *International Symposium on Wavelets Theory & Applications in Applied Mathematics, Signal Processing & Modern Science (WAV '08),* Istanbul, Turkey, May 27-30, 2008

[10] Kekre, H. B., Sudeep D. Thepade, and Akshay Maloo. "Performance Comparison of Image Retrieval Using Fractional Coefficients of Transformed Image Using DCT, Walsh, Haar and Kekre's Transform." *CSC-International Journal of Image processing (IJIP) 4.2 (2010): 142-155.*
[11] Hu, Ming-Kuei. "Visual pattern recognition by moment invariants." Information Theory, *IRE Transactions on 8.2 (1962): 179-187*.
[12] M.M. Mokji, S.A.R. Abu Bakar, "Gray Level Co-occurrence Matrix Computation Based on Haar Wavelet", *Computer graphics, Imaging and Visualization(CGIV) IEEE,*2007
[13] LI Pingxiang and FANG Shenghui, "SAR Image Classification Based on its Texture Features", *Geo Spatial Information Sience Volume-6, Issue-3,* September 2003
[14] http://www.scanex.ru/imgs/data-quickbird-sample3.jpg