IMAGE RETRIEVAL USING CO-OCCURRENCE MATRIX & TEXTON CO-OCCURRENCE MATRIX FOR HIGH PERFORMANCE

Sunita P. Aware
Department of IT., Jawaharlal Darda C.O.E., Yavatmal (MS), India

ABSTRACT

This paper put forward a new method of co-occurrence matrix to describe image features. In this paper putting a new implemented work which is comparison with texton co-occurrence matrix to describe image features. Maximum work done successfully using texton co-occurrence matrix. A new class of texture features based on the co-occurrence of gray levels at points. These features are compared with previous types of co-occurrence based features, and experimental results are presented indicating that the new features should be useful for texture. The results demonstrate that it is much more efficient than representative image feature descriptors, such as the auto-correlogram and the texton co-occurrence and the texton co-occurrence matrix. It has good discrimination power of texture features.

KEYWORDS: *Image retrieval, gray level co-occurrence matrix, wavelet transform texton co-occurrence matrix.*

I. INTRODUCTION

Texture is one of the most commonly used features used to analyze and interpret images, specifically medical images. Texture is a measure of the variation of the intensity of a surface, quantifying properties such as smoothness, coarseness, and regularity. it is often used as a region descriptor in image analysis and computer vision. Specifically, a textured region consists of a connected set of pixels that satisfy a given gray-level property which occurs repeatedly in an image region [1]. Several methods have been applied towards the analysis and characterization of texture within medical images including fractal dimension, run-length encoding, discrete wavelet transform, and two-dimensional co-occurrence matrices [2]. Of those mentioned, in texture analysis, two-dimensional dependence matrices [co-occurrence matrices] are extensively used; they are able to capture the spatial dependence of gray-levels which contributes to perception of texture [1].

In this paper, we investigate a new approach to the co-occurrence matrix currently used to extract textural features co-occurrence matrices. At this point, it is critical to make a clear distinction between our proposed approach and that of 3D co-occurrence matrices.

As it is presented in the literature, 3D co-occurrence matrices are calculated by summing pixel pair probabilities in a 2D image, as opposed to the pixel pair probabilities that are summed in 2D co-occurrence matrices [5]. In earlier days of machine based image retrieval, images were first annotated with text and then searched using a text based approach. This manual annotation was a cumbersome and expensive. As a result, it is difficult for the traditional text-based method to retrieval of images from database. In order to resolve this problem, a new technique known as content based image retrieval (CBIR) evolved. CBIR is a technique which uses visual contents of an image such as color, shape and texture to search images from large image database. This technique finds application in advertising, medicine, crime detection, entertainment, and digital libraries.

II. METHODOLOGY

A. Description of co-occurrence matrix:

Suppose an image to be analyzed is rectangular and has Nx. resolution cells in the horizontal direction and Ny resolution cells in the vertical direction. Suppose that the gray tone appearing in each resolution cell is quantized to Ng levels. Let Lx = {1,2,……,Nx} be the horizontal spatial domain, Ly = {1,2,…….,Ny} be the vertical spatial domain, and G = {1,2, …,Ng } be the set of Ng quantized gray tones. The set Ly X Lx is the set of resolution cells of the image ordered by their row-column designations.[3]. The image I can be represented as a function which assigns some gray tone in G to each resolution cell or pair of coordinates in Ly X Lx; I: Ly X Lx =>G.[2]. An essential component of our conceptual framework of texture is a measure, or more precisely, four closely related measures from which all of our texture features are derived. These measures are arrays termed angular nearest-neighbor gray-tone spatial-dependence matrices, and to describe these arrays we must emphasize our notion of adjacent or nearest-neighbour resolution cells themselves. [4]. We consider a resolution cell-excluding those on the periphery of an image, etc.to have eight nearest-neighbor resolution cells.[1].
We assume that the texture-context information in an image I is contained in the overall or "average" spatial relationship which the gray tones in image I have to one another. More specifically, we shall assume that this texture context information is adequately specified by the matrix of relative frequencies Pij with which two neighboring resolution cells separated by distance „d‟ occur on the image, one with gray tone i and the other with gray tone j. Such matrices of gray-tone spatial-dependence frequencies are a function of the angular relationship between the neighboring resolution cells as well as a function of the distance between them.[6]. The set of all horizontal neighboring resolution cells separated by distance 1. This set, along with the image gray tones, would be used to calculate a distance 1 horizontal gray-tone spatial-dependence matrix. Formally, for angles quantized to 450 intervals the un normalized frequencies are defined as follows…

P(i,j,d,00) = #{((k ,l),(m,n)) € (Ly x Lx) (1)
x (Ly x Lx) |k-m|= 0,|1 –n| =d,
I(k,l) = i, I(m,n) = il

P(i,j,d,450) = #{((k ,l),(m,n)) € (Ly x Lx) (2)
x (Ly x Lx)I(k-m = d, l -n =-d)
Or (k - m = -d, l - n =d),
I (k, l) = i, I (m, n) = j

P(i,j,d,900) = #{((k,l),(m,n)) € (Ly x Lx) (3)
x (Ly x Lx) | k- m| = d,|l-n|=0
I(k,l)=i,I(m,n)=j}

P(i,j,d,1350) = #{((k,l),(m,n)) e (Ly x Lx) (4)
x (Ly x Lx) (k - m = d, l- n = d)
Or (k-m = -d, l-n= -d),
I (k, l) = i, I(m,n) = j} (1)

Where # denotes the number of elements in the set.
Note that these matrices are symmetric;
P (i, j; d, a) = P (j, i; d, a). (5)
The distance metric p implicit in the preceding equations can be explicitly defined by
P ((k, l), (m, n)) = max {|k – m|, |1- n|}. (6)

B. Grey-level co-occurrence matrix texture:

Grey-Level Co-occurrence Matrix texture measurements have been the workhorse of image texture since they were proposed by Haralick in the 1970s. To many image analysts, they are a button you push in the software that yields a band whose use improves classification – or not. [11].
The original works are necessarily condensed and mathematical, making the process difficult to understand for the student or front-line image analyst.
Calculate the selected Feature. This calculation uses only the values in the GLCM. See:

i) Contrast
ii) Correlation
iii) Energy
iv) Homogeneity
These features are calculated with distance 1 and angle 0, 45 and 90 degrees. 2.3 K-Means Clustering [6].
A cluster is a collection of data objects that are similar to one another with in the same cluster and are dissimilar to the objects in the other clusters. It is the best suited for data mining because of its efficiency in processing large data sets. It is defined as follows:
The k-means algorithm is built upon four basic operations:
a. Selection of the initial k-means for k-clusters.
b. Calculation of the dissimilarity between an object and the mean of a cluster.
c. Allocation of an object of the cluster whose mean is nearest to the object.
d. Re-calculation of the mean of a cluster from the object allocated to it so that the intra cluster dissimilarity is minimized. [4].
The advantage of K-means algorithm is that it works well when clusters are not well separated from each other, which is frequently encountered in images.

C. Textural Features extracted from co-occurrence matrices:

Our initial assumption in characterizing image texture is that all the texture information is contained in the gray-tone spatial-dependence matrices. Hence all the textural features we suggest are extracted from these gray-tone spatial-dependence matrices. Some of these measures relate to specific textural characteristics of the image such as homogeneity, contrast, and the presence of organized structure within the image [3]. Other measures characterize the complexity and nature of gray tone transitions which occur in the image. Even though these features contain information about the textural characteristics of the image, it is hard to identify which specific textural characteristic is represented by each of these features. For illustrative purposes, we will define 3 of the 14 textural features in this section and explain the significance of these features in terms of the kind of values they take on for two images of distinctly different textural characteristics the features we consider are as follows

D. Textural Features:

a. Angular Second Moment:
b. Contrast:
c. Correlation:
d. Sum of Squares: Variance:
e. Inverse Difference Moment:
f. Sum Average:
g. Sum Variance:
h. Sum Entropy:
i. Entropy:
j. Difference Variance:
k. Difference Entropy:
l. Information Measures of Correlation:
Usually the neighboring pixels in an image are not very distinct (i.e. they are highly correlated). Quite often in an image there are large regions of pixels with nearly the same color, such as the sky, or with uniform texture, such as walls, cloth, or sand. For typical pixel-level texture feature extraction, the texture values for each pixel are computed with the sliding window positioned such that the pixel is the center of the window. Clearly, for images of the type mentioned above, the neighboring pixels have the same (or nearly the same) texture features.[5]. In such cases, nearly identical results are generated by performing nearly identical computations. If we can determine beforehand which computations will result in nearly identical results, we can avoid these calculations, trading off decreased computational complexity with a small amount of distortion in the texture extraction results.
Features for the representative pixel of the block. Otherwise, we examine its sub-blocks in a similar way. Hence, each pixel will get its texture features in one of these blocks at a certain hierarchical level, either by copying the corresponding representative pixel' s texture features, or by computing its own texture features if it is a representative pixel itself.

Backbone blocks and key pixels (out-of-block pixels are not included), when a = 2, b= 4
The whole image can be divided into many non-overlapping b*b pixel blocks (b = 2a, aÎN). We call these blocks 1st level backbone blocks, denoted B1. Each B1 can be divided into 4 b/2*b/2-sized 2nd level backbone blocks, B2. Similarly, each B2 can be divided into 4 b/4*b/4-sized 3rd level backbone blocks, and so on. Each block can be divided further and further until each backbone block contains only one pixel. [6]. These single-pixel backbone blocks are (a+1)th level backbone blocks.
For each backbone block Bs, where 1 <= s <= (a+1), the pixel at its upper-left corner is called its key pixel, denoted Pkey(Bs). If p(i,j) is a key pixel of a sth level backbone block, we call this backbone block Bs(i, j), i.e. Pkey(Bs(i, j)) = p(i,j). Backbone blocks and key pixels are shown in Figure Notice that the key pixels of larger backbone blocks are always the key pixels of smaller backbone blocks.
If the image dimensions, M and N, are not multiples of b, the size of the 1st level backbone block, there will be some pixels that do not res in 1st level backbone blocks.[1]. We call these pixels out-of-block pixels, which we will examine a little later. Except for out-of-block pixels, each pixelA resides in one of the 1st level backbone blocks. Since each higher level backbone block is generated by dividing the next lower level block into 4 equal-sized sub-blocks, each pixel must also be in one backbone block at each level. For example, for a pixel p(i,j), at the sth level, where 1 <= s <= (a+1), it is in backbone block
Bs(_i / 2s+a-1 _ *2s+a-1, _ _ j / 2s+a-1 *2s+a-1), (7)
and the corresponding key pixel for this backbone block is
p(_ _ i / 2s+a-1 *2s+a-1, _ _ j / 2s+a- 1 *2s+a-1). (8)
Then, each pixel will get its texture features in one of these backbone blocks, either by copying the corresponding key pixel' s texture features, or by computing its own texture features if it is a key pixel itself. If we let key pixel p(i,j) be the upper-left pixel of a k*k sliding window to which the wavelet transform is applied, we can associate the texture feature extracted by this transform window to p(i,j). We use VTK h(p(i,j)), VTK v(p(i,j)), and VTK o(p(i,j)) to denote the texture features for key pixel p(i,j), namely, the key pixel texture features, in the horizontal, vertical and oblique directions, respectively. Observe that this sliding window is the same as the sliding window for extracting texture feature for pixel p(i+k/2, j+k/2) which is described in section 1, and shown in Figure 3, i.e.

VTK h(p(i,j))= Vt h(p(i+k/2, j+k/2)) , (9)
VTKv(p(i,j))= Vt v(p(i+k/2, j+k/2)), (10)
VTK o(p(i,j))= Vto(p(i+k/2, j+k/2)). (11)

In this paper, we let k = b = 2a. So,
VTK h(p(i,j))= Vt h(p(i+2a-1, j+2a-1)) , (12)
VTK v(p(i,j))= Vt v(p(i+2a-1, j+2a-1)), (13)
VTKo(p(i,j))= Vt o(p(i+2a-1, j+2a-1)). (14)

Note, this assignment causes a misalignment between the texture values and their associated pixels, so our algorithm realigns them after assigning all texture values, as discussed below.
For a backbone block Bs, if we let its key pixel Pkey(Bs) be the upper-left pixel of the sliding window for the wavelet transform, and let the size of the sliding window be 2s+a-1, i.e. the sliding window covers the backbone block exactly, we can associate the texture feature extracted by this transform window to Bs. We use VTB h(Bs), VTB v(Bs), VTB o(Bs) to denote those texture features, namely, the backbone block texture features, in the horizontal, vertical and oblique directions, respectively. Observe that this sliding window is the same as the sliding window for extracting key pixel texture feature[4].

III. CO-OCCURRENCE MATRIX

A. Measurements of color gradients and edge detection:

Comparison with previous algorithm which is wavelet base algorithm see the result of this algorithm.
First create data base entries.
1. These are seven entries.
2. We can see the texture feature of image
3. Following see the texture feature of last image.

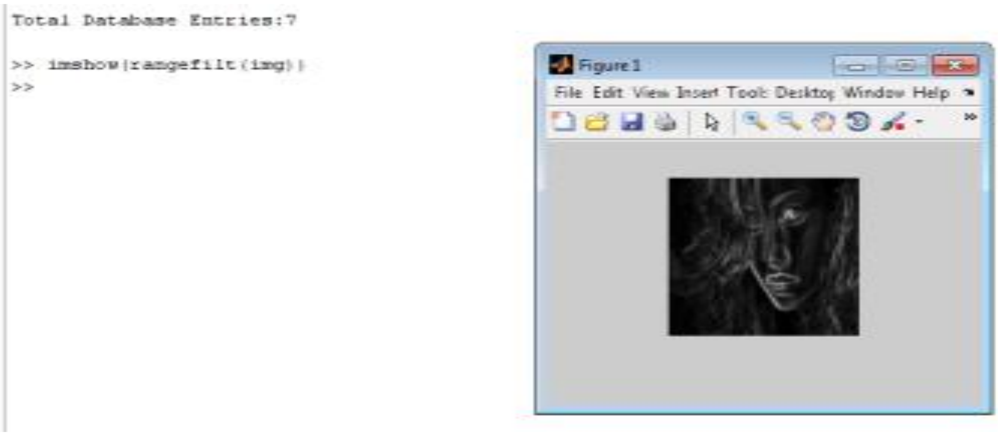

Fig. 1 Texture Teature of Image of Data base

1. Wavelet base.
2. That will be match image retrieval.
3. you will have given How many percentage accurcy?

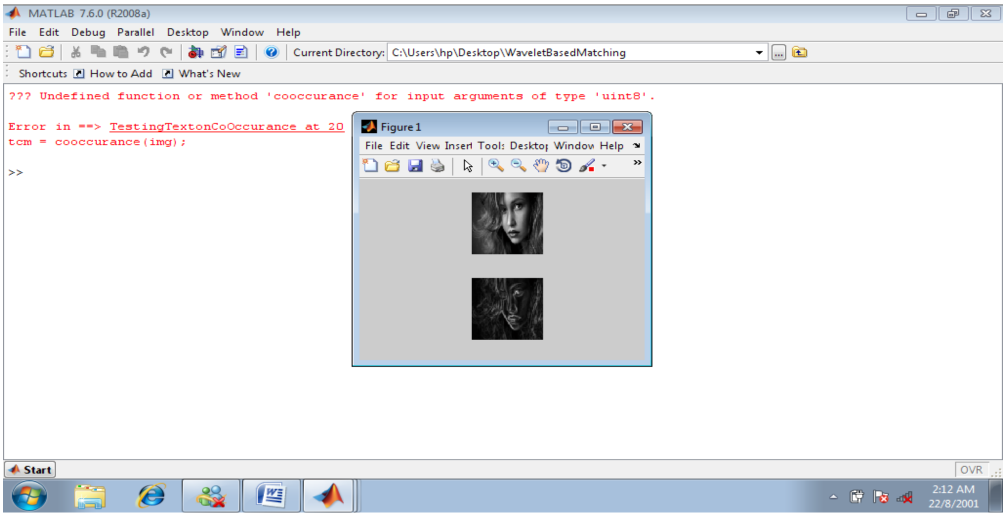

Fig. 2 Output of Wavelet Transform on Input Image

1. This result is compare with image retrieval
2. It gives accuracy 83.63%

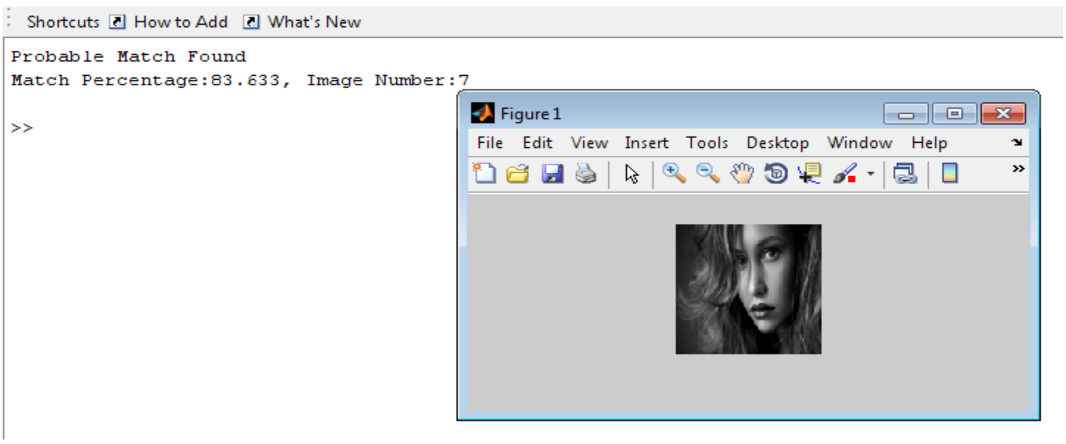

Fig. 3 Output of Image Retrieval

1. In this result shows the remaining % of noise present in this algorithm shows the noisy image of this algorithm

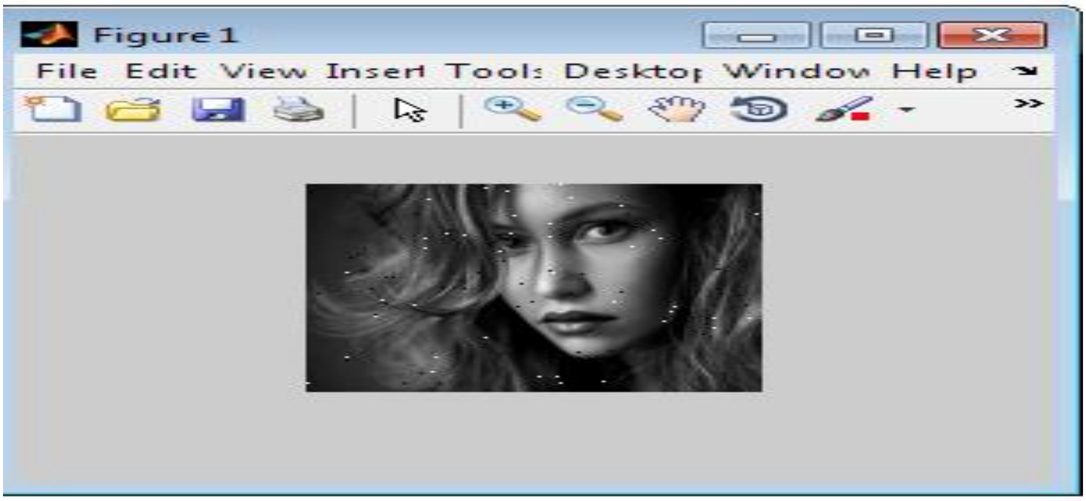

Fig.4 Output of Co-occurrence matrix

IV. TEXTON CO-OCCURRENCE MATRIX

1. This algorithm is used for this project.
2. How is it best?
3. Most important advantages is image retrieval base
4. First you will get created data base see the following
5. Here created in database 24 entries and last entry shows texture feature of image

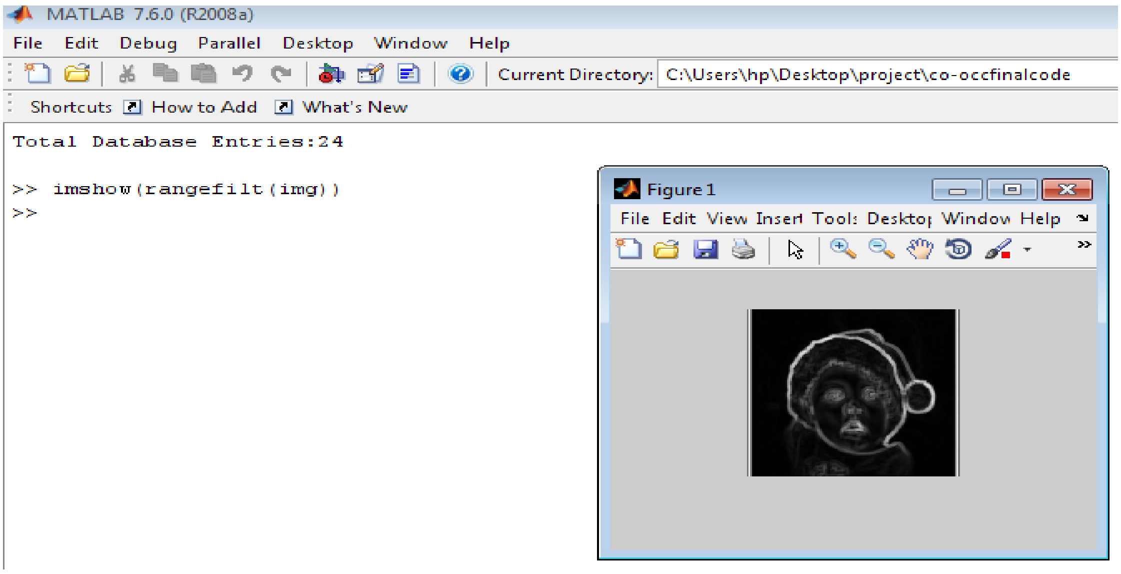

Fig. 5 Texture Image of Database

1. you will get Texton co-occurrence matrix image
2. We will match this Texton co-occurrence matrix with database

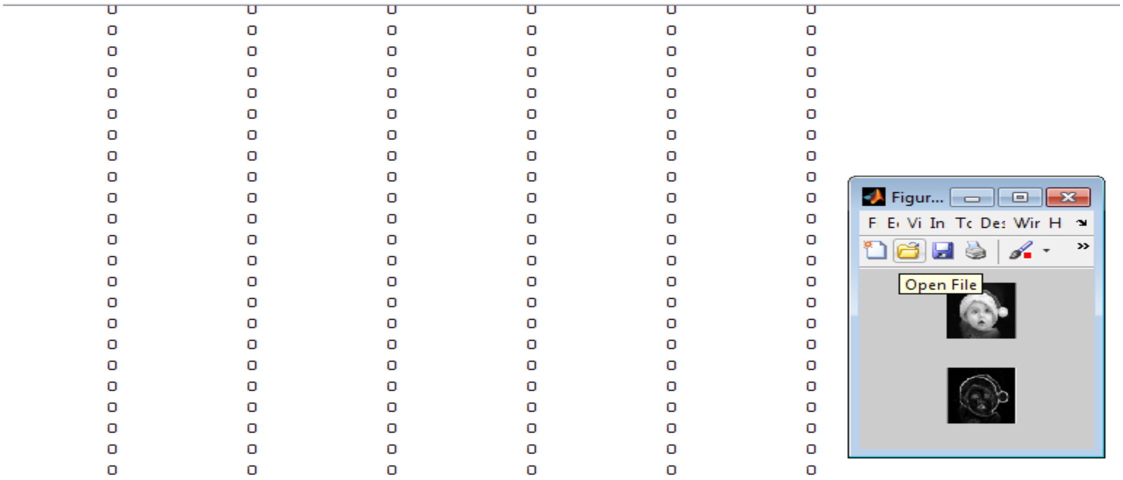

Fig. 6 Image with Texton Co-occurrence Matrix

Database Loaded
Finding Texon Co-occurance of Input Image
Texon Co-occurance found for the image
Image 1, Match Percent 39.35
Image 2, Match Percent 52.96
Image 3, Match Percent 40.55
Image 4, Match Percent 48.69
Image 5, Match Percent 48.69
Image 6, Match Percent 48.22
Image 7, Match Percent 42.15
Image 8, Match Percent 50.90
Image 9, Match Percent 60.55
Image 10, Match Percent 60.23
Image 11, Match Percent 44.42
Image 12, Match Percent 34.08
Image 13, Match Percent 34.35
Image 14, Match Percent 66.68
Image 15, Match Percent 52.66
Image 16, Match Percent 66.68
Image 17, Match Percent 49.00
Image 18, Match Percent 49.92
Image 19, Match Percent 45.00
Image 20, Match Percent 45.54
Image 21, Match Percent 46.09
Image 22, Match Percent 64.16
Image 23, Match Percent 47.09
Image 24, Match Percent 95.11
Probable Match Found
Match Percentage:95.111, Image Number:24

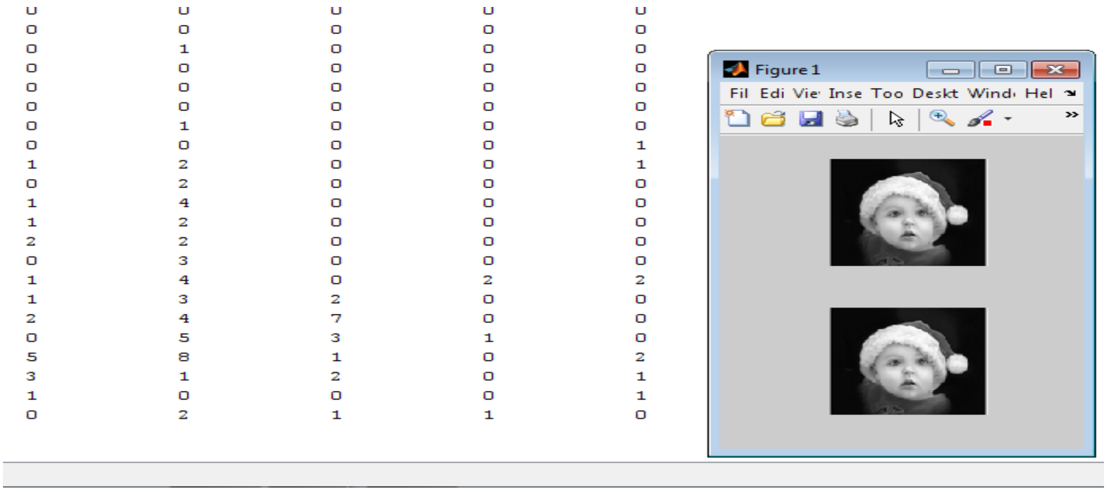

Fig. 7 Output of Texton Co-occurrence Matrix Match with Image Retrieval 95.11%

1. Output of Texton co-occurrence matrix match with image retrieval 95.111%

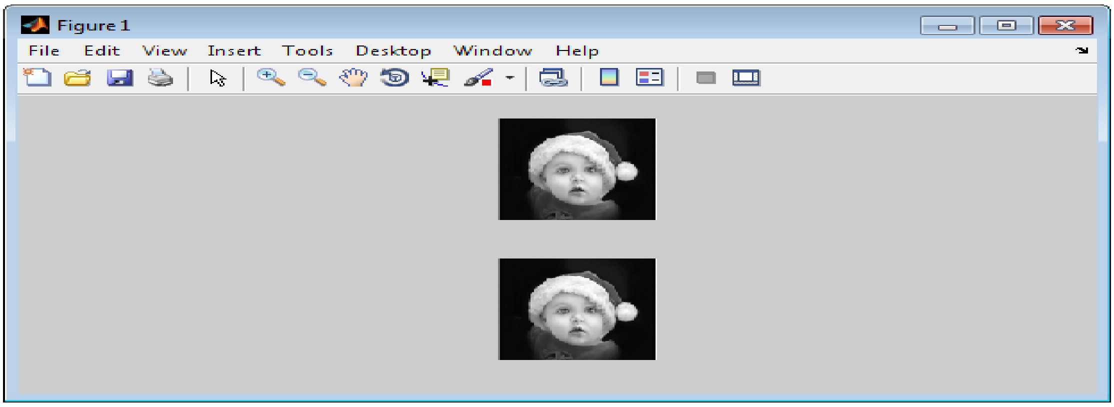

Fig. 8 Output of Image of Co-occurrence Matrix with Texton Co-occurrence matrix

V. Application

A. Textural Features extracted from co-occurrence matrices:

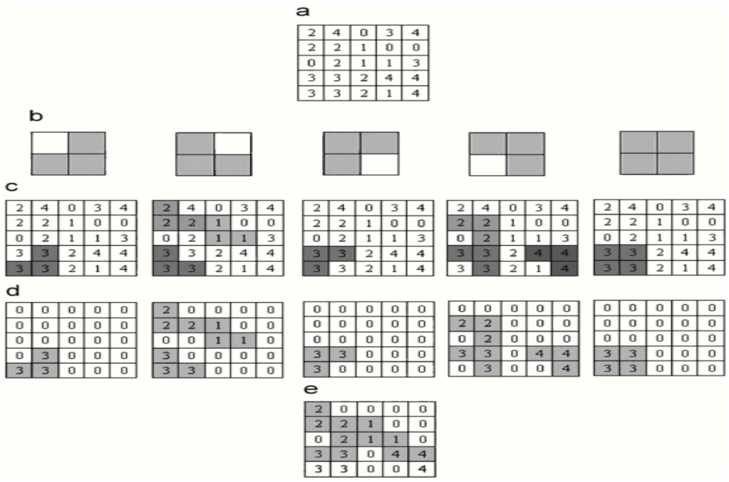

Fig.9 The flow chart of textons detecting: (a) original image, (b) five special types of textons, (c) textons detection (d) five components of texton images and (e) the final texton image.

Our initial assumption in characterizing image texture is that all the texture information is contained in the gray-tone spatial-dependence matrices. Hence all the textural features we suggest are extracted from these gray-tone spatial-dependence matrices. Some of these measures relate to specific textural characteristics of the image such as homogeneity, contrast, and the presence of organized structure within the image. Other measures characterize the complexity and nature of gray tone transitions which occur in the image.[4]. Even though these features contain information about the textural characteristics of the image, it is hard to identify which specific textural characteristic is represented by each of these features.

B. Calculation of texton co-occurrence matrix:

Julesz proposed the term "texton" conceptually more than 20 years ago.[4,5] Texton is a very useful concept in texture analysis. As a general rule, texton defined as a set of blobs or emergent patterns sharing a common property all over the image; however, defining textons remains a challenge. [11]

The image features have a close relationship with textons and color diversification. The difference of textons may form various image Features. If the textons in image are small and the tonal differences between neighboring textons are large, a fine texture may result. If the textons are larger and consist of several pixels, a coarse texture may result. At the same time, the fine or coarse texture characteristic depends on scale. If the textons in image are large and consist of a coarse texture characteristic depends on scale. If the textons in image are large and consist of a few texton categories, an obvious shape may result few texton categories, an obvious shape may result.

There are many types of textons in images [6,9]. In this paper, we only define five special types of textons for image analysis. Let there is a 2×2 grid in image. Its pixels are V1, V2, V3and V4 if three or four pixel values special types of textons are denoted as are same, thus those pixels formed a texton. Those five T1,T2,T3,T4,and T5 It is shown in Fig. 4, the shadow of 2×2 grid denotes those pixel values are same. Different shadow structure formed various textons is an image, if it is shifted by one pixel in every direction, a 2×2 grid may appear.[7]

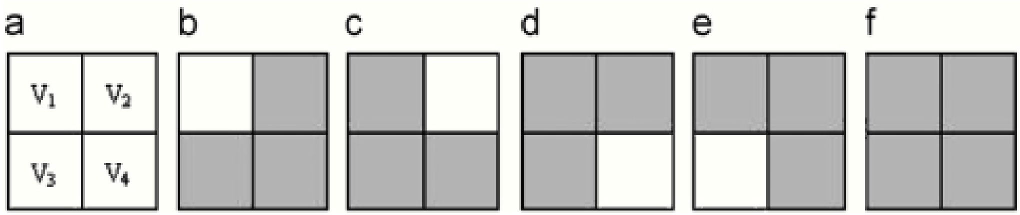

Fig. 10 Five special types of texto ns: (a) 2 ×2 grid; (b) T1; (c) T2;(d) T3;(e)T4;(f) and T5.

We use those five special types of textons to detect every grid, respectively, and then find out whether one of them may appear in those grids. A type of texton may detect out a component of texton image [13], thus there are five components of texton images. It is shown in Fig5(c). In those five components of texton images, the pixels of textons are kept in original values, others are replaced with the value of 0. It is shown in Fig. 5(d). Finally, we will combine those five components of texton images together to form a final textons image. Let the pixel position is P = (X, Y), at the same position Pi = (x,y), every component of texton image has a pixel value, thus five components of texton images have five pixel values. They are denoted as W1, W2, W3, W4 and W5 If those five pixel values are same, the final texton image will be kept in original, values in corresponding positions. If the values of 0 and nonzero appear in those five pixels, the final textons image will be kept in the values of nonzero.

VI. SYSTEM REQUIREMENTS

The experiments were performed on a single CPU 2.8GHz Pentium PC with 512MB memory and the Windows operating system. The average time consuming of the gray co-occurrence matrix, the color correlograms and the TCM are 28.04 ms, 27.54ms and 125.78 ms.

VII. RESULT

Performance Curve & Observation Tables:

In experiments, the distance parameter *D* values utilized to calculate the co-occurrence matrixes were: *D* = 1, 2…. 9. The average retrieval precision values are listed in Tables 1 and 2. The best performance of TCM was obtained when *D* = 5 for the first database set and *D*=5 for the second database set. The average retrieval precision of TCM method is basically described from 59.81% to 59.7% for the first database set and from for the second database set. The average retrieval precision of DWT method is from 48.6% to 51.2% for the first database set and from 59% to 59.2% for the second database set. The average retrieval precision and recall curves are plotted in Fig. 6.4. It can be seen from the Tables 1, 2, 3, and Fig. 6.4 that the proposed method achieves good results in terms of the retrieval precisions and recall compared to DWT methods. Its performance is much better than that of DWT and TCM methods.

On the first image database set, the average retrieval precision of TCM method beyond DWT method is 42.22 % to 42.5%, respectively. On the second image database set, it is beyond DWT and TCM method, 47.5 % and 61.95.08%, respectively

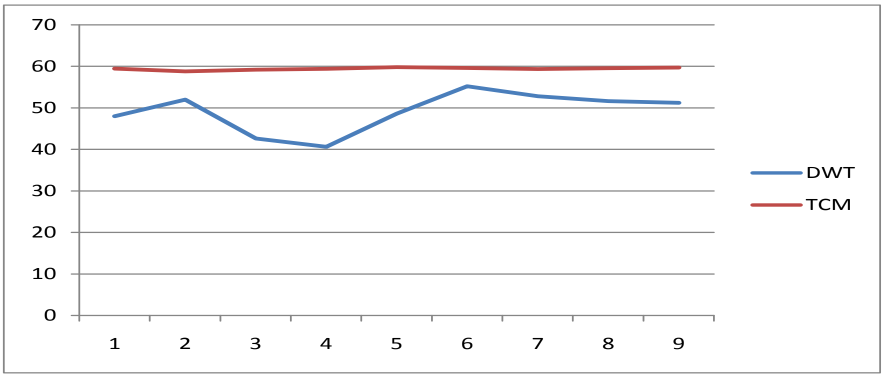

Table1: The average retrival precision with different distance on the first image database set

Methods	Distance parameter and the average retrieval precision								
	D=1	D=2	D=3	D=4	D=5	D=6	D=7	D=8	D=9
DWT	48	52	42.6	40.63	48.6	55.2	52.8	51.6	51.2
TCM	59.47	58.79	59.19	59.42	59.81	59.61	59.33	59.56	59.7

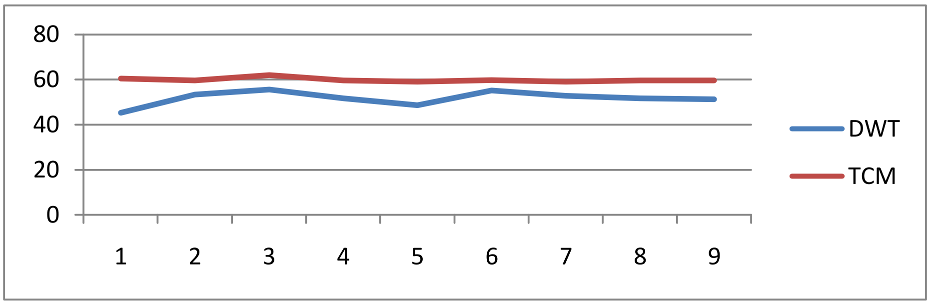

Table2 : The Average Retrieval precision on the second image database se

Methods	(Distance parameter and the average retrieval precision)								
	D=1	D=2	D=3	D=4	D=5	D=6	D=7	D=8	D=9
DWT	45.2	53.4	55.6	51.6	48.6	55.2	52.8	51.6	51.2
TCM	60.39	59.61	61.95	59.56	59	59.66	59	59.51	59.5

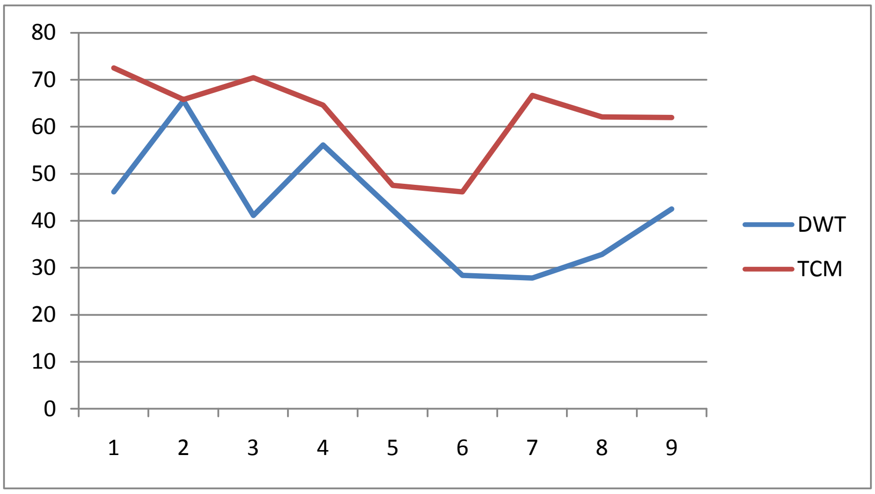

Table3: The average retrieval precision with different distance on the third image database set

Methods	(Distance parameter and the average retrieval precision)								
	D=1	D=2	D=3	D=4	D=5	D=6	D=7	D=8	D=9
DWT	46.11	65.55	41.11	56.11	42.22	28.33	27.78	32.78	42.5
TCM	72.5	65.75	70.42	64.58	47.5	46.11	66.67	62.08	61.95

VIII. FUTURE WORK

In many applications, the evaluation of image retrieval by people can only be made subjectively ,but the human beings' eyes to distinguish the texture roughness capacity is limited, in other words, low roughness of certain small changes cannot be aware. This vision of how to adapt to, first of all we think of a variety of features integrated. However, a different image database, because of its different content, the next step is to work against the image of the contents and the fractal dimension of the application of the start.

IX. CONCLUSION

1. In this paper, we have put forward a new method of co occurrence matrix to describe image features. It is different from the gray co-occurrence matrix and color correlograms because this method has the discrimination power of shape features. Image retrieval experiments were conducted over two image database sets using the gray co-occurrence matrix, color correlograms and the texton co-occurrence matrix. The two image database sets mainly come from VisTex texture database of MIT, Corel images and web. Experimental results have shown that our method has the discrimination power of color, texture and shape features, the performances are better than that of GLCM and CCG. Above shown the results of current algorithm which is Texton Co-occurrence Matrix. Current algorithm shows the texture feature & match with the image retrieval.
2. Current algorithm compare with previous algorithm which is wavelet base algorithm.
3. In previous algorithm near about 10 % to 20 % noise is there.
4. Due to noise speed is slow than the current algorithm.
5. It will take more time than the current algorithm.
6. Accuracy is less than current algorithm.
7. Current algorithm is more efficient than previous algorithm.
8. Current algorithm is more than 95% match with the image retrieval.
9. Above results shown the difference between Wavelate base algorithm and Texton co-occurrence matrix.

REFERENCES

[1]. Guang-Hai Liu, Jing-Yu Yang., Image retrieval based on the texton co-occurrence matrix in IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[2]. R.M. Haralick, K. Shangmugam, I. Dinstein, Textural feature for image classification, IEEE Trans. Syst. Man Cybern. SMC-3 (6) (1973) 610–621.
[3]. G. Cross, A. Jain, Markov random field texture models, IEEE Trans. Pattern Anal Mach. Intell. 5 (1) (1983) 25–39.
[4]. J. Mao, A. Jain, Texture classification and segmentation using multi-resolution simultaneous autoregressive models, Pattern Recognition 25 (2) (1992) 173–188.
[5]. F. Liu, R. Picard, Periodicity, directionality, and randomness: wold features for image modeling and retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (7) (1996) 722–733.
[6]. B.S. Manjunath, W.Y. Ma, Texture features for browsing and retrieval of image data, IEEE Trans. Pattern Anal. Mach. Intell. 18 (8) (1996) 837–842.
[7]. J. Han, K.-K. Ma, Rotation-invariant and scale-invariant Gabor features for texture image retrieval, Image Vision Comput. 25 (2007) 1474–1481.
[8]. T. Chang, C.C. Jay Kuo, Texture analysis and classification with tree-structured wavelet transform, IEEE Trans. Image Process. 2 (4) (1993) 429–441.
[9]. A. Laine, J. Fan, Texture classification by wavelet packet signatures, IEEE Trans. Pattern Anal. Mach. Intell. 11 (15) (1993) 1186–1191.
[10]. M. Sonka, V. Hlavac, R. Boyle, Image Processing, Analysis, and Machine Vision, second ed., Thomson Brooks/Cole, Boston, MA, USA, 1998.
[11]. C. Palm, Color texture classification by integrative co-occurrence matrices, Pattern Recognition 5 (37) (2004) 965–976.
[12] J. Mao, A. Jain, Texture classification and segmentation using multi-resolution simultaneous autoregressive models, Pattern Recognition 25 (2) (1992) 173–188.
[13] F. Liu, R. Picard, Periodicity, directionality, and randomness: wold features for image modeling and retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 18 (7) (1996) 722–733.

Author

Sunita P. Aware Assist. Prof. in IT Department at JDIET, Yavatmal. Completed B.E.in EXTC branch from Aurangabad University & M.Tech in VLSI from Nagpur University having seven years teaching experience