IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-2, Issue-11 E-ISSN: 2347-2693

Content Based Image Retrieval Using Extended Local Tetra Patterns

Aasish Sipani[1*], Phani Krishna[2] and Sarath Chandra[3]

[1*, 2, 3] *Department of Computer Science and Engineering, National Institute of Technology, Warangal, India,*
aasishsipani@gmail.com, ppkk1992@gmail.com, chandra.3162@gmail.com

www.ijcaonline.org

Received: Oct/26/2014 Revised: Nov/09/2014 Accepted: Nov/20/2014 Published: Nov/30/2014

Abstract— In this modern world, finding the desired image from huge databases has been a vital problem. Content Based Image Retrieval is an efficient method to do this. Many texture based CBIR methods have been proposed so far for better and efficient image retrieval. We aim to give a better image retrieval method by extending the Local Tetra Patterns (LTrP) for CBIR using texture classification by using additional features like Moment Invariants and Color moments. These features give additional information about the color and rotational invariance. So an improvement in the efficiency of image retrieval using CBIR is expected.

Keywords— Content Based Image Retrieval (CBIR), Local Tetra Patterns (LTrP), Gabor Filters, Histogram Equalization, Moment Invariants

I. INTRODUCTION

The explosive growth of digital libraries due to Web cameras, digital cameras, and mobile phones equipped with such devices is making the database management by human annotation an extremely tedious and clumsy task. Thus, there exists a need for developing an efficient approach to search for the desired images form huge databases. Content Based Image Retrieval (CBIR) is one of the adopted solutions to the problem.

Content Based Image Retrieval (CBIR) has become an important area for people to search and retrieve information. CBIR system retrieves the relevant images from the image data base for the given query image, by comparing the feature of the query image and images in the database. The CBIR utilizes visual contents of an image such as color, texture, shape, faces, spatial layout, etc., to represent and index the image database. These features can be further classified as general features such as color, texture, and shape, and domain-specific features such as human faces, fingerprints, etc.

Texture Analysis has been one of the prominent topics in today's world which is used in computer vision and pattern recognition. Texture Analysis is broadly justified into texture classification and texture segmentation. Texture classification is used to determine to which of a finite number of physically defined classes a homogeneous texture region belongs. The Texture Classification is done by using Discrete Wavelet Transform (DWT). However, the DWT can extract only three-directional (Horizontal, vertical, and diagonal) information from an image. To address this directional limitation, Gabor transform (GT) is proposed for texture image retrieval

Corresponding Author: *Aasish Sipani*

The concept of local descriptors is one of the methods of statistical way of texture classification. Many Local Descriptors have been proposed to date like:

i) Local Binary Patterns (LBP)
ii) Local Derivative Patterns (LDP)
iii) Local Ternary Patterns (LTP) and
iv) Local Ternary Patterns (LTrP)

These local descriptors use the information of the neighboring pixels to form patterns which can be further helpful for image retrieval.

In our proposed system of CBIR, the already proposed Local Tetra Patterns (LTrP) has been extended by adding Color moments and moment invariants as the feature vectors for Image retrieval. The Color features can be obtained by histogram processing of the R, G & B Colors obtained from the images in the Database. The moment invariants are the moments which do not vary with translation, rotation and resizing of the same image. These features are significant to retrieve the images which are transformed. All these features combined with the Gabor filtered Local Tetra Patterns is expected to give a much better Retrieval of similar images which is the objective of CBIR.

The Local Binary Patterns (LBP) [1] was the first kind of local pattern proposed for texture feature extraction. It became a great success in the field of texture classification and retrieval. Various extensions of the LBP have been proposed which are proved to be greater success than the normal LBP. The LBP operator on facial expression analysis and recognition is successfully reported by Xi Li *et al.* Various extensions of Local Binary Patterns like LBP variance with global matching, Dominant LBPs, Completed

LBPs, joint distribution of local patterns with Gaussian mixture etc. have been proposed for rotational invariant texture classification.

Zhang *et al.* proposed Local Derivative Patterns (LDPs) [1] for face recognition, where they considered the LBP as non-directional first-order local patterns collected from the first-order derivatives and extended the same approach for nth order LDPs. This was the first local pattern descriptor which considered the derivatives for texture classification.

The versions of the LBP and the LDP in the open literature cannot adequately deal with the range of appearance variations that commonly occur in unconstrained natural images due to illumination, pose, facial expression, aging, partial occlusions, etc. In order to address this problem, the Local Ternary Pattern (LTP) [4] has been introduced for face recognition under different lighting conditions. The Local Ternary Patterns use the concept of thresholds for obtaining the information from the patterns.

The LBP, the LDP, and the LTP extract the information based on the distribution of edges, which are coded using only two directions (positive direction or negative direction). This drawback motivated in the development of a new pattern descriptor called Local Tetra Patterns (LTrP) [2]. In LTrP, along with the magnitude and intensity of the neighboring pixels, also focuses on the relationship between the direction of the central pixel and its neighboring pixels. The LTrP is able to encode images with four distinct values as it is able to extract more detailed information than LBP, LTP and LDP. The LBP and the LTP encode the relationship between the gray value of the center pixel and its neighbors, whereas the LTrP encodes the relationship between the center pixel and its neighbors based on directions that are calculated with the help of (n-1) th-order derivatives. This meant that more information about the neighboring pixels resulted in a better texture classification and therefore a better CBIR. This is evident from the fact that in LBP, only the neighbors are considered for forming the patterns. In LDP, even their derivatives are considered while from LTP, two different LBPs can be drawn out, which in turn suggests that it is nearly twice as efficient as LBP

The LTrP is able to encode the images using four distinct values and the computational cost of it is less when compared to the other pattern descriptors. In the LTrP, the horizontal and vertical pixels of the neighbors have been used for derivative calculation. This resulted in a total of 13 Local Binary Patterns (LBPs) from a single LTrP. So it is 13 times more informative than the LBP. But in all these patterns, only textural information has been used. The use of other features can improve the efficiency of CBIR. This inspired us to extend the LTrP along with Color Moments and the moment invariants as the added features for an effective CBIR.

II. PROCEDURE

The model proposed [2] as shown in Fig 1consists of two phases namely Feature Extraction Phase and Similarity Measurement phase. A feature is something which describes an image. Thousands of features can be extracted from a single image. These features extracted are known as feature vectors. The first phase is the extraction of these feature vectors from the images. All the images in the Database are pre-processed to extract the required features. The features extracted in the proposed method are texture, color moments and moment invariants.
The images in the database and the query image are processed so as to extract feature vectors from them. Now, in order to retrieve the similar images from the numerous images in the Database, there should be some similarity measures. This stage is the second stage of image retrieval. In this stage, each feature vector of all the images in the database is compared with the feature vector of the query image to find out the similarity between the images. The more similar images have less distance i.e. less dissimilarities.
The output of the second stage of the proposed system is the image retrieval. The similar images are retrieved from the Database. The performance of the system can be measured by Average Precision, Average Recall and Efficiency.

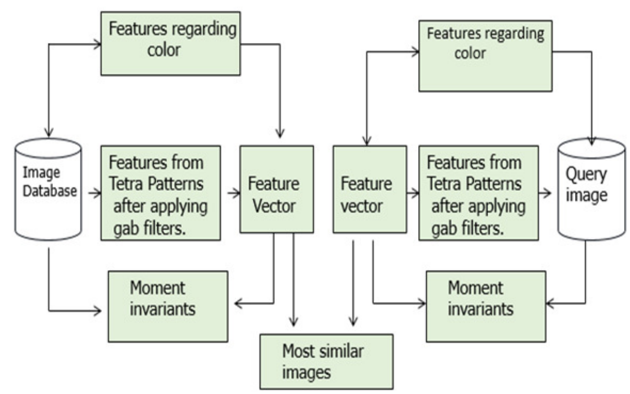

Fig 1. Architectural diagram of the proposed system

III. DESIGN AND ANALYSIS

We have proposed to phases of the system:

1.) Feature Extraction Stage
2.) Similarity Measurement Stage

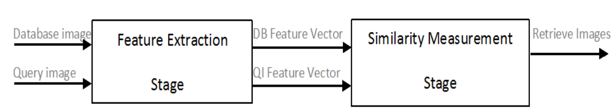

Fig 2. Phases of the System

A. *Feature Extraction Stage*

The various features obtained from the images in the proposed CBIR System are based on Texture, Color and Moment Invariants. All these three features combined, is expected to give a better image retrieval and therefore a better CBIR system.

Each and every color image is a combination of 3 components namely Red (R), Green (G) and Blue (B). The color image is segregated into these 3 components and features are extracted from each component.

Histogram equalization is a method in image processing of contrast adjustment using the image's histogram. This method increases the global contrast of the images, especially when the usable data has close contrast values. This allows for areas of lower local contrast to gain a higher contrast.

The histogram equalization is an approach to enhance a given image. The approach is to design a transformation *T (.)* such that the gray values in the output are uniformly distributed in [0, 1]. The result of the histogram equalized image is that the dark pixels appear darker and the bright pixels appear brighter. This allows for a better contrast in the image which in turn gives a better image.

The statistical data like mean, variance and skewness are obtained from the equalized histograms. A total of 9 color features are obtained.

So far, in all the pattern descriptors, only the pixel values of the neighbors have been considered so as to form the pattern. Local Tetra Pattern is a new local descriptor which encodes the relationship between the referenced pixel and its neighbors based on directions.

Given an image I, the first-order derivatives along 0 and 90 directions are denoted as

$I^1_{\theta}(g_p)|_{\theta=0°, 90°}$. Let g_c denote the center pixel in I, Let g_h and g_v denote the horizontal and vertical neighborhoods of g_c, respectively. Then the first order derivatives [2] at the center pixel g_c can be defined as:

$$I^1_{\theta}(g_c)|_{\theta=0^\circ} = I(g_h) - I(g_c) \qquad (1)$$

$$I^1_{\theta}(g_c)|_{\theta=0^\circ} = I(g_v) - I(g_c) \qquad (2)$$

The direction of the neighborhood pixel is calculated as follows [2]:

$$I^1_{Dir}(g_c) = \begin{cases} 1, & I^1_{0^o}(g_c) \geq 0 \text{ and } I^1_{90^o}(g_c) \geq 0 \\ 2, & I^1_{0^o}(g_c) < 0 \text{ and } I^1_{90^o}(g_c) \geq 0 \\ 3, & I^1_{0^o}(g_c) < 0 \text{ and } I^1_{90^o}(g_c) < 0 \\ 4, & I^1_{0^o}(g_c) \geq 0 \text{ and } I^1_{90^o}(g_c) < 0 \end{cases} \qquad (3)$$

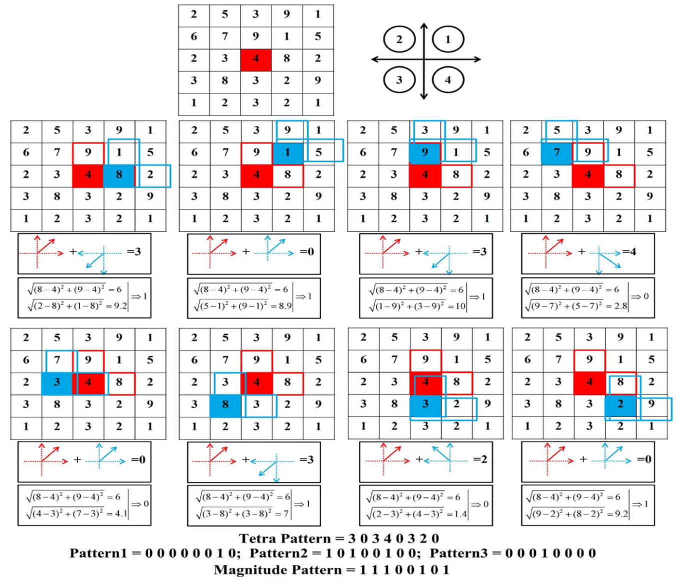

Fig 3. Local Tetra patterns and Magnitude Pattern

The figure 3 [2] is an Example to obtain the tetra and magnitude patterns. For generating a tetra pattern, the bit is coded with the direction of neighbour when the direction of the center pixel and its neighbour are different, otherwise "0." For the magnitude pattern, the bit is coded with "1" when the magnitude of the center pixel is less than the magnitude of its neighbour, otherwise "0."

When we apply first-order derivative in horizontal and vertical directions to the neighbourhood pixel "8," we obtain direction "3" and magnitude "9.2." Since the direction of the center pixel and the direction obtained from the neighbourhood pixel are not same, we assign value "3" to the corresponding bit of the LTrP. It can be seen that the magnitude of the center pixel is "6," which is less than the magnitude of neighbourhood pixel. Hence, we assign value "1" to the corresponding bit of the magnitude pattern. Similarly, the remaining bits of the LTrP and the magnitude pattern for the other seven neighbours are computed resulting in the tetra pattern "3 0 3 4 0 3 2 0" and the magnitude binary pattern "1 1 1 0 0 1 0 1."After coding the tetra pattern, we separate it into three binary patterns as follows. Referring to the generated LTrP, the first pattern is obtained by keeping "1" where the tetra pattern value is "2" and "0" for other values, i.e., "0 0 0 0 0 0 1 0." Similarly, the other two binary patterns "1 0 1 0 0 1 0 0" and "0 0 0 1 0 0 0 0" are computed for tetra pattern values "3" and "4," respectively.

For each of the 12 patterns, a histogram is obtained. All these histograms are processed so as to get the statistical data from the images. The statistical data includes mean, variance and skewness. The same procedure is followed for the images in the database and then for the query image and all the values are stored in separate databases.

Since there are 12 histograms in total, a total of 36 features are obtained from this step. These 36 features are further compared for image retrieval.

Gabor filters are the most widely used filters in image processing and especially in Content Based Image Retrieval. A set of Gabor filters with different frequencies and orientations may be helpful for extracting useful features from an image. Gabor filters have been widely used in pattern analysis applications.

The Gabor filter is applied on different orientations to remove noise in all the possible orientations. Applying Gabor Filter on each image in 0 and 90 directions [1], we get

$$G^{m-1}_{m,0^\circ}(g_c) = G^{n-2}_{m,0^\circ}(g_c) - G^{n-2}_{m,0^\circ}(g_h) m = 1,2,3 \quad (4)$$

$$G^{n-1}_{m,90^\circ}(g_c) = G^{n-2}_{m,90^\circ}(g_c) - G^{n-2}_{m,90^\circ}(g_v) m = 1,2,3 \quad (5)$$

$$I^1_{0^\circ}(g_c) => G^{n-1}_{m,0^\circ}(g_c) \text{ and } I^1_{90^\circ}(g_c) => G^{n-1}_{m,90^\circ}(g_h) \quad (6)$$

When we apply the Gabor filter in different orientations and then the Gabor filter is convoluted with the images so as to obtain the Gabor Filtered image. LTrP patterns are obtained from this image where the direction is calculated as shown in Fig. From these patterns, again, the 36 statistical values are obtained which are used for image retrieval.

There are a total of 7 moments which do not change with translation, rotation, resizing and mirroring of an image. These 7 moments are used for feature extraction step for Content Based Image Retrieval.

B. Similarity Measurment Stage

The output of the first stage is the features which are extracted from the images. The values of feature vectors of the query image and the images in database are calculated and stored separately.

The feature vector for the query image Q represented as $f_Q = (f_{Q_1}, f_{Q_2}, \ldots\ldots f_{Q_{L_q}})$ is obtained from feature extraction. Similarly, each image in the database is represented with the feature vector $f_{DB_j} = \left(f_{DB_{j1}}, f_{DB_{j2}}, \ldots\ldots f_{DB_{jL_q}}\right)$ j=1,2,.....|DB|.

The goal is to select the best images that resemble the query image. This involves the selection of top-matched images by measuring the distance between the query image and the images in database |DB|. In order to match the images, we use d_1 [4] similarity distance metric computed using

$$\sum_{i=1}^{L_g} \left| \frac{f_{DB_{ji}} - f_{Q_i}}{1 + f_{DB_{ji}} + f_{Q_i}} \right| \quad (7)$$

Where $f_{DB_{ji}}$ is the i^{th} feature of the j^{th} image in database |DB|.

The images with the smallest distance are said to be more similar to the query image. N most similar images are retrieved from the database and efficiency of the system is obtained by using standard performance measures like Average Precision, Average Recall an Average Retrieval Rate.

IV. ALGORITHMS

A. Algorithm for finding the Local Tetra Pattern [2]

1. The derivatives of the center pixel d_c in the horizontal and vertical directions are calculated using

$$I^1_\theta(g_c)|_{\theta=0^\circ} = I(g_h) - I(g_c)$$

$$I^1_\theta(g_c)|_{\theta=0^\circ} = I(g_v) - I(g_c)$$

2. The direction of the center pixel is defines as follows:

$$I^1_{Dir}(g_c) = \begin{cases} 1, & I^1_{0^o}(g_c) \geq 0 \text{ and } I^1_{90^o}(g_c) \geq 0 \\ 2, & I^1_{0^o}(g_c) < 0 \text{ and } I^1_{90^o}(g_c) \geq 0 \\ 3, & I^1_{0^o}(g_c) < 0 \text{ and } I^1_{90^o}(g_c) < 0 \\ 4, & I^1_{0^o}(g_c) \geq 0 \text{ and } I^1_{90^o}(g_c) < 0 \end{cases}$$

3. Calculate the direction d_n of the neighboring pixels of g_c using Step 2 by taking each neighborhood pixel as the center pixel.
4. If the direction of the center pixel matches with direction of the neighboring pixel, the pattern due to that neighboring pixel is 0 else it is d_n
5. Repeat step 4 for all the 8 neighbors of a center pixel
6. An 8 bit Local Tetra Pattern is obtained

This 8 bit Local Tetra Pattern is divided into 3 Local Binary Patterns of 8 bits each and histograms of these patterns are obtained

B. Gabor Feature Extraction

1. Obtain the images in the database
2. Prepare Gabor filters with 0 and 90 orientations
3. The images in the database are convoluted with the Gabor filters obtained in step 2 to obtain Gabor filtered images [5]

$$FGab(x, y, W, \theta, \sigma_x, \sigma_y) = \sum_k \sum_l F(x-k, y-l) * Gab(x, y, W, \theta, \sigma_x, \sigma_y)$$

4. Calculate mean from Gabor filtered image

$$\mu(W, \theta, \sigma_x, \sigma_y) = \frac{1}{XY} \sum_{x=1}^{X} \sum_{y=1}^{Y} FGab(x, y, W, \theta, \sigma_x, \sigma_y)$$

5. Calculate standard deviation from the Gabor filtered image

$$std(W,\theta,\sigma_x,\sigma_y) = \sqrt{\sum_{x=1}^{X}\sum_{y=1}^{Y} ||FGab(x,y,W,\theta,\sigma_x,\sigma_y)| - \mu(W,\theta,\sigma_x,\sigma_y)|^2}$$

6. Calculate skewness for the Gabor filtered image

$$Skew = \frac{1}{XY} \times \sum_{x=1}^{X}\sum_{y=1}^{Y}\left(\frac{FGab(x,y,W,\theta,\sigma_x,\sigma_y)| - \mu(W,\theta,\sigma_x,\sigma_y)}{std(W,\theta,\sigma_x,\sigma_y)}\right)^3$$

7. Repeat steps 2 to 6 on the query image.

The results obtained are compared for similarity measurement to retrieve images similar to query image.

C. *Color Feature Extraction*

1. Obtain the images in the database.
2. For each image, separate the Red, Green and Blue components of the images
3. Obtain the histograms for each component of image separately.
4. Normalize the histograms
5. Calculate mean of the normalized image
$$M_k^1 = \frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y} f_k(x,y)$$
6. Calculate standard deviation of the normalized image
$$M_k^2 = \left(\frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}(f_k(x,y) - M_k^1)^2\right)^{\frac{1}{2}}$$
7. Calculate skewness of the normalized image
$$M_k^3 = \left(\frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}(f_k(x,y) - M_k^1)^3\right)^{\frac{1}{3}}$$
8. Repeat steps 2 to 7 for the query image

Compare the results for obtaining similarity measures.

D. *Moment Invariants*

The 7 moments are calculated using the following formulae [5]

- $\emptyset_1 = \mu_{20} + \mu_{02}$
- $\emptyset_2 = [\mu_{20} - \mu_{02}]^2 + 4\mu_{11}^2$
- $\emptyset_3 = [\mu_{30} - 3\mu_{02}]^2 + [3\mu_{21} - \mu_{03}]^2$
- $\emptyset_4 = [\mu_{30} + 3\mu_{12}]^2 + [\mu_{21} + \mu_{03}]^2$
- $\emptyset_5 = [\mu_{30} - 3\mu_{12}][\mu_{30} + \mu_{12}] \times [(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] + [3\mu_{21} - \mu_{03}][\mu_{21} + \mu_{03}] \times [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]$
- $\emptyset_6 = [\mu_{20} - \mu_{02}][(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2] + 4\mu_{11}[\mu_{30} + \mu_{12}][\mu_{21} + \mu_{03}]$
- $\emptyset_7 = [3\mu_{21} - \mu_{03}][\mu_{30} + \mu_{12}] \times [(\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2] - -[\mu_{03} - 3\mu_{12}][\mu_{21} + \mu_{03}] \times [3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2]$

Where

- $m_{pq} = \sum_{x=1}^{X}\sum_{y=1}^{Y} x^p y^q f(x,y)$
- $m_{pq} = \sum_{x=1}^{X}\sum_{y=1}^{Y} (x - I)^p (y - J)^q f(x,y)$
- $m_{pq} = \iint F_{pq}(x,y)\, f(x,y)dxdy$
- $\mu_{pq} = \frac{m_{pq}}{(m_{00})\beta}, \beta = \frac{p+q}{2} + 1$

The 7 moments obtained will remain the same even if the image is translated or rotated or resizing or after mirroring.

E. *Proposed Sytem Framework [2]*

Input: Query image Q, Database DB

Output: Retrieval Result

1. Load the image, and convert it into grayscale.
2. Apply the first-order derivatives in horizontal and vertical axis.
3. Calculate the direction for every pixel.
4. Divide the patterns into four parts based on the direction of the center pixel.
5. Calculate the tetra patterns, and separate them into three patterns.
6. Calculate the histograms of binary patterns.

7. Obtain the individual Color components of the Color image.
8. Obtain the histograms for individual components.
9. Equalize the histogram obtained in step 8
10. Combine the histograms calculated from steps 6 and 8.
11. Construct the feature vector.
12. Compare the query image with the images in the database using distance measures
13. Retrieve the images based on the best matches.

V. RESULTS

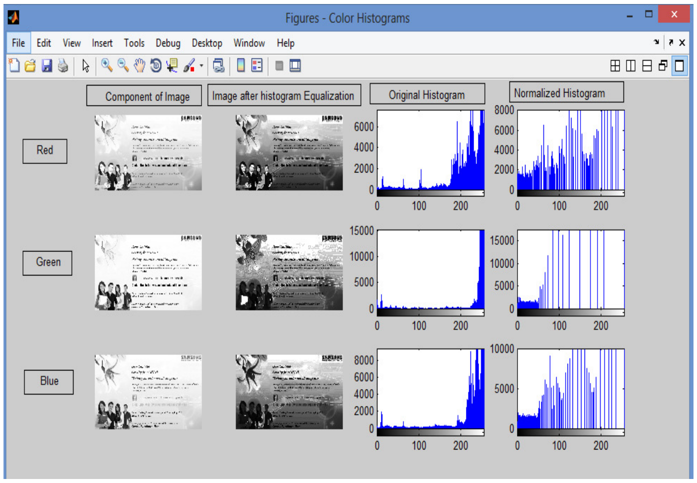

Fig 4. Color Histograms

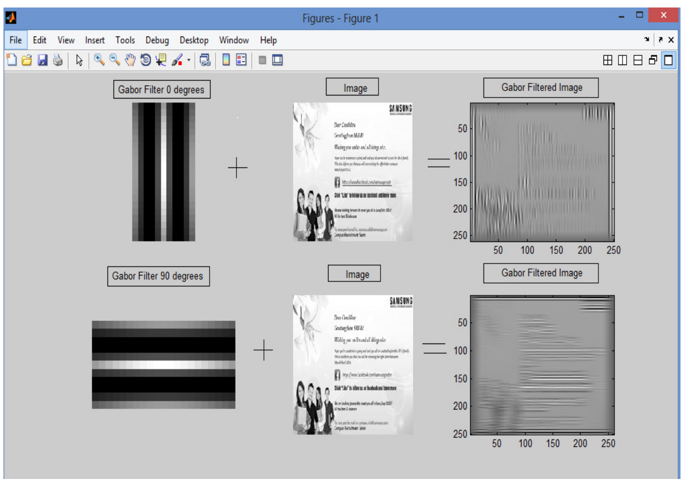

Fig 5. Gabor filtered images

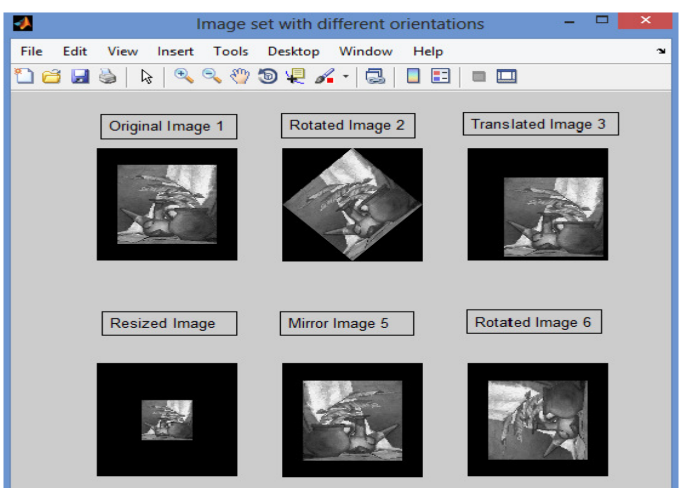

Fig 6. Moment Invariant Table Set

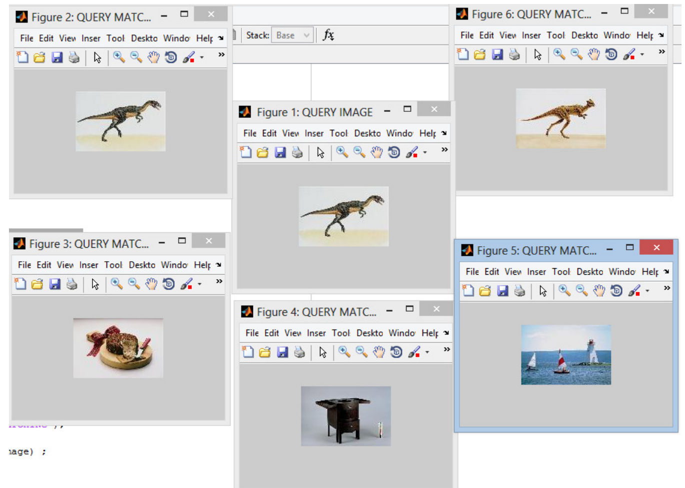

Fig 7. LTrP Results

Given a query image of dinosaur, the first 5 images retrieved contained only 2 images of dinosaurs.

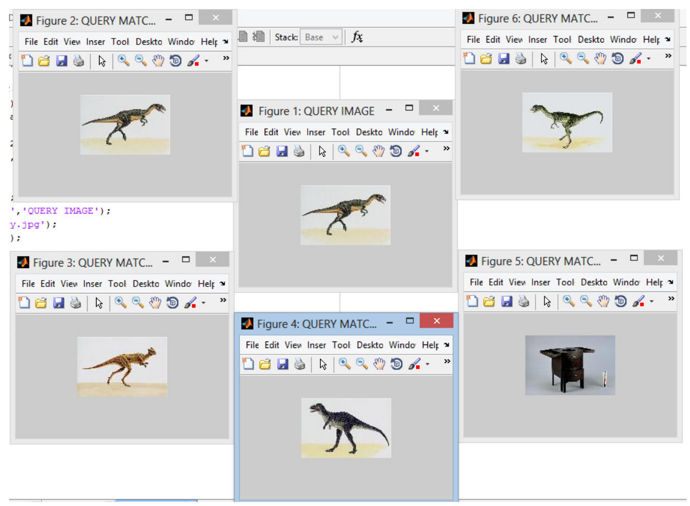

Fig 8. Proposed CBIR Results

Given a query image of dinosaur, out of the first 5 images retrieved, 4 images are similar to dinosaurs.

VI. CONCLUSION

We have taken 10 classifications of 100 images each from Corel DB. For precision, top 50 images are considered. Proposed CBIR with and without skewness and LTrP systems have been compared based on the efficiency, average precision and average recall.

Method	Efficiency (in %)	Avg. Precision (in %)	Avg. Recall (in %)
Proposed CBIR with Skewness	75.74	73.05	37.127
Proposed CBIR without Skewness	58.12	61.71	31.125
LTrP	65.48	64.86	32.865

Fig 9. Final results for comparison

- The average precision has improved from 64.86% in LTrP to 73.05% in our proposed system of CBIR.
- The average recall has improved from 32.865% in LTrP to 37.127% in our proposed system of CBIR.
- The overall efficiency has improved from 65.48% in LTrP to 75.74% in our proposed system of CBIR.

ACKNOWLEDGMENT

We wish to express our gratitude to our respected guide Dr. USN Raju, Assistant Professor, Computer Science and Engineering, National Institute of Technology, Warangal for his valuable guidance in our thesis. We are indebted to him for helping us at every stage for its successful completion.

REFERENCES

[1] Baochang Zhang, Yongsheng Gao, Sanqiang Zhao, and Jianzhuang Liu, "Local Derivative Pattern Versus Local Binary Pattern: Face Recognition With High- Order Local Pattern Descriptor" in IEEE Transactions on Image Processing, Vol. 19, No. 2, pp 533 - 544 Feb. 2010.

[2] Subrahmanyam Murala, R. P. Maheshwari and R. Balasubramanian, "Local Tetra Patterns: A New Feature Descriptor for Content-Based Image Retrieval" in IEEE Transactions on Image Processing, Vol. 21, No. 5, pp 2874 - 2886 May. 2012.

[3] S. Liao, Max W. K. Law, and Albert C. S. Chung "Dominant Local Binary Patterns for Texture Classification" in IEEE Transactions on Image Processing, Vol. 18, No. 5, pp 1107 – 1118 May. 2009.

[4] P.V. N. Reddy and K. Satya Prasad, "Inter Color Local Ternary Patterns for Image Indexing and Retrieval", DOI: DIP102011005, CiiT International Journal for Digital Image Processing, Oct.2011.

[5] T. A. Mitchell, R. Bowden & M. Sarhadi, "Efficient texture analysis for industrial inspection**",** International Journal of Production Research, 38:4, 967-984, DOI: 10.1080/002075400189248, Nov.2010.

[6] Abdenour Hadid, "The Local Binary Pattern Approach and its Applications to Face Analysis", Image Processing, Theory, Tools and Applications, Conference of Sousse, pp 1-9 Nov.2010

