IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)

# Analysis of Random Forest Classifier based Attribute Filtering for Query based Image Retrieval

Neha Malav
*M. Tech. Scholar*
Rajasthan College of Engineering for Women, Jaipur (India)
neha_malav30@rediffmail.com

Rakesh Sharma
*Asst. Professor*
Rajasthan College of Engineering for Women, Jaipur (India)
mtech@rcew.ac.in

*Abstract* –**Content-based image retrieval (CBIR) is the application of computer vision to the image retrieval problem, that is, the problem of searching for digital images in large databases. Most of the existing image retrieval systems give lesser accuracy while using different features individually. Therefore, in this research work we develop a framework using extraction of color, texture and shape features along with Random Forest Classifier to classify the extracted features. For similarity measurement, we have used Euclidian distances for each feature. Performance of the proposed research work is carried out using certain evaluation parameters, namely; False Negative Rate, False Positive Rate, True Positive Rate and True Negative Rate.**

*Keywords* – **CBIR, Color Features, Euclidian distance, Texture features, False Negative Rate, False Positive Rate, Random Forest Classifier, Shape Features, True Negative Rate and True Positive Rate.**

## I. Introduction

The area of image retrieval has been a dynamic exploration range for a few decades and has been given careful consideration lately as a consequence of the dramatic and quick increment in the use of digital images. The improvement of Internet not just cause a violently increasing volume of digital images, additionally give individuals more approaches to get those images. The vitality of a viable strategy in searching and retrieving images from huge collection can't be overemphasized. One methodology for indexing and recovering image information is utilizing manual content annotations. The annotations can then be utilized to search images by implication. Yet there are a few issues with this methodology. Initially, it is extremely hard to portray the contents of an image utilizing just a couple of keywords. Second, the manual annotation procedure is exceptionally subjective, uncertain, and deficient. Those issues have made incredible requests for automatic and compelling procedures for content-based image retrieval (CBIR) frameworks. Most CBIR frameworks utilize low-level image features, for example, color, texture, shape, edge, and so on, for image indexing and retrieval. This is on account of the low-level features can be processed automatically.

The thought behind content-based retrieval is to recover, from a database, media things, (for example, images, audio and video) that are pertinent to a given query. Relevancy is judged on the basis of the content of media items. A few steps are required for this. To start with, the features from the media items are extracted and their values and indices are spared in the database. At that point the index structure is utilized to preferably filter out all immaterial things by checking attributes with the client's query. Finally, attributes of the correlated items are compared with the help of some similarity check to the features of the query and recovered items are situated in place of likeness.

Image databases and accumulations can be huge in size, containing hundreds, thousands or even a large number of images. The ordinary strategy for image retrieval is looking for an essential keyword that would match the enlightening keyword relegated to the image by a human categorizer [1]. Right now being worked on, despite the fact that few frameworks exist, is the retrieval of images focused around their content, called Content Based Image Retrieval, CBIR. While computationally lavish, the results have more accuracy than traditional image indexing. Thus, there exists a trade-off in the middle of accuracy and computational expense. This trade-off diminishes as more proficient algorithms are used and expanded computational power gets to be economical.

## II. Content Based Image Retrieval

CBIR or Content Based Image Retrieval is the retrieval of images based on visual features such as colour, texture and shape [2]. Reasons for its development are that in many large image databases, traditional methods of image indexing have proven to be insufficient, laborious, and extremely time consuming. These old methods of image indexing, ranging from storing an image in the database and

IJDACR
ISSN: 2319-4863
IJDACR
International Journal Of Digital Application & Contemporary Research
International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)

associating it with a keyword or number, to associating it with a categorized description, have become obsolete. This is not CBIR. In CBIR, every image that is put away in the database has its features extraction and contrasted with the features of the query image. It includes two phases [3]:

- ***Feature Extraction:*** The initial phase in this technique is feature extraction of image to a distinct degree.
- ***Matching:*** The second phase includes matching these features to yield a result that is outwardly comparative.

***Qualities of Image Queries***

What sorts of queries are clients liable to put to an image database?

To answer this inquiry inside and out obliges a definite learning of client needs:

- Why clients look for images?
- What use they make of them?
- How they judge the utility of the images they recover?

As we realize that, insufficient exploration has yet been accounted for to answer these inquiries. Even a single technical individual suggest that still images are needed for a variety of objects, containing:

- With the help of content articles, it is hard to express feelings.
- Showing thorough information for examination.
- Formal recording of outline information for later utilization.

Access to a required image from an archive may accordingly include a search for images delineating particular sorts of object, inspiring a particular disposition, or essentially contain a particular pattern or texture. Images have many forms of attribute which could be used for image retrieval, are as fallow:

- The presence of a particular combination of colour, texture or shape features (e.g. green stars)
- The presence or arrangement of specific types of object (e.g. chairs around a table)
- The depiction of a particular type of event (e.g. a football match)
- The presence of named individuals, locations, or events (e.g. the Queen greeting a crowd)
- Subjective emotions one might associate with the image (e.g. happiness)
- Metadata like: who created the image, where and when?

Each one recorded query sort (except for the last) speaks to a more elevated amount of reflection than its ancestor, and each one is harder to reply without reference to some assortment of outer information.

Basically there are three levels of queries:

**First Level:** It involves recovery by primitive feature, for example, color, texture, shape or the spatial area of picture features. Occurrences of such inquiries may incorporate, "discover me more pictures that resemble this", "discover pictures with long thin dull objects in the upper left-hand corner". This level of image retrieval uses topographies which are objective, and directly derivable from the images, without any external knowledge. Its use is largely limited to specialist applications like: trademark registration, colour matching of fashion accessories or documentation of drawings in a design archive.

**Second Level:** It embodies recovery by determined feature, containing some level of logical derivation about the character of the objects delineated in the picture. It can practically be divided as:

- Retrieval of objects of a given type (e.g. "find pictures of a double-decker bus")
- Retrieval of particular objects or individuals ("discover a picture of the Eiffel tower").

To answer queries at this level, reference to certain outside store of information is generally required – mainly for the more specific queries at level 2(b). In the first occurrence over, some former comprehension is important to recognize an item as a transport instead of a lorry; in the second example, one needs the data that a given individual structure has been given the assignment "the Eiffel tower". Examine rules at this level, mostly at level 2(b), are generally still practically objective. This level of query is more than level 1

**Third Level:** It comprises retrieval by abstract attributes, covers a weighty sum of high-level reasoning around the meaning and resolution of the objects or scenes depicted. This level of retrieval can be subdivided into:

- Retrieval of named events or types of activity (e.g. "find pictures of Scottish folk dancing")
- Retrieval of pictures with passionate or religious implication ("discover a picture portraying enduring").

Achievement in noting queries at this level can oblige some complexity from the explorer. Subjective judgment and Complex thinking, can be obliged to make the connection between content of picture and the theoretical ideas it is obliged to exhibit.

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

**International Journal of Digital Application & Contemporary research**
**Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)**

Research and development subjects in CBIR cover a variety of areas, several shared with conventional image processing and information retrieval. Certain of the most vital are:

- Understanding image user's necessity and information-seeking manners.
- Documentation of suitable ways of describing image content.
- Mining features from raw images.
- Providing compacted storage for bulky image databases.
- Matching the query and put away pictures in a manner that reflects human comparability judgements.
- Efficiently retrieving stored images by content.
- Providing usable human interfaces to CBIR model.

***Applications of CBIR***

Cases of CBIR applications are [4]:

*In Prevention of Crime:* Automatic face authentication frameworks, utilized by police department.

*Security Check:* Retina scanning or Finger print for authentication.

*Diagnosis in the Medical Field:* Using CBIR in a medicinal database of medical images to support finding by distinguishing comparative previous severe cases.

*Legal Possessions*: Registration of trademark picture, where another candidate imprint is contrasted with existing imprints with guarantee no danger of confounding property possession.

## III. Proposed Methodology

Figure 1 exhibits the proposed approach used in this paper:

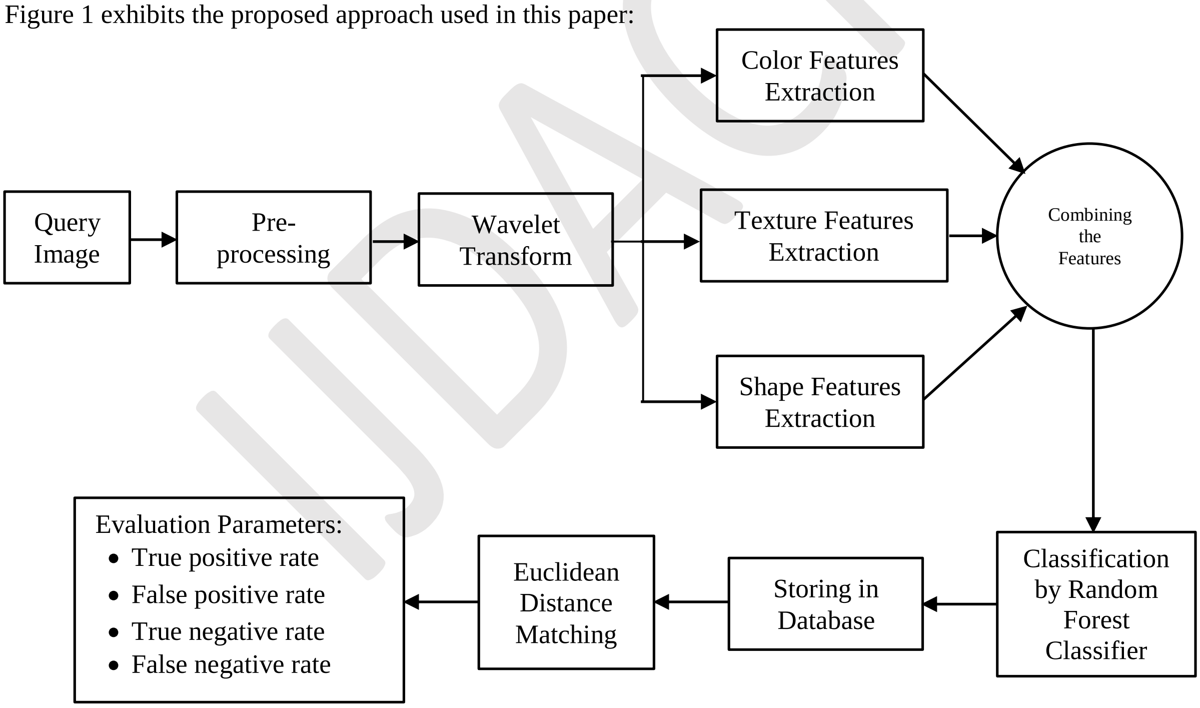


Figure 1: Basic block diagram for proposed research work

This paper intends a novel content based image retrieval framework utilizing color, texture and shape features. Initially, a query image is selected arbitrarily from the database. Since the database contains images of different dimensions. To extract certain features from the image, it is necessary to reduce the dimensions of the selected image. Pre-processing is applied to lessen the dimension of selected image trailed by discrete wavelet transform. The next step is to extract certain features from image which are as follows:

**Color Features Extraction**

1. RGB to HSV Conversion

An image pixel value is converted from the RGB representation using the formula shown below:

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

**International Journal of Digital Application & Contemporary research**
**Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)**

$$H = \cos^{-1}\frac{\frac{1}{2}[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}} \quad (1)$$

$$S = 1 - \frac{3[\min(R,G,B)]}{R+G+B} \quad (2)$$

$$V = \left[\frac{R+G+B}{3}\right] \quad (3)$$

2. Color Moments
   a. Mean
   b. Standard Deviation

If the value of the $i^{th}$ color channel at the $j^{th}$ image pixel is $I_{ij}$ and the number of pixels is $N$, then the index entries related to this color channel and the region '$r$' are known as the color moments followed by the formula:

***Mean:***

$$E_{r,i} = \frac{1}{N}\sum_{j=1}^{N} I_{ij} \quad (4)$$

***Standard Deviation:***

$$\sigma_{r,i} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}\left(I_{ij} - E_{r,i}\right)^2} \quad (5)$$

Where $E_{r,i}$ $(1 \leq i \leq 3)$ represents the average color (mean) of the region $r$ and $\sigma_{r,i}$ represents the standard deviation of the region $r$. And the extracted color features are given by the following feature vector:

$$f_c = \{E_{1,1}, \sigma_{1,1} E_{2,2}, \sigma_{2,2} E_{3,3}, \sigma_{3,3} \ldots\ldots\ldots\ldots\ldots. E_{r,i}, \sigma_{r,i}\} \quad (6)$$

**Texture Features Extraction**

For a given image $I(x,y)$ having size $P \times Q$, the discrete Gabor wavelet transform is expressed by the following convolution:

$$G_{mn}(x,y) = \sum_s \sum_t I\,(x-s, y-t)\psi_{mn}^*(s,t) \quad (7)$$

Where, $s$ and $t$ are the filter mask size variables, and $\psi_{mn}^*$ is the complex conjugate of $\psi_{mn}$.

$$\psi(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)\right].exp(j2\pi Wx) \quad (8)$$

Where $W$ denotes the modulation frequency. Now the generating function:

$$\psi_{mn}(x,y) = a^{-m}\psi(\tilde{x},\tilde{y}) \quad (9)$$

Where $m$ and $n$ specify the scale and orientation of the wavelet respectively.

And $m = 0,1,\ldots M-1, n = 0,1,\ldots,N-1$, and

$$\tilde{x} = a^{-m}(x\cos\theta + y\sin\theta) \quad (10)$$

$$\tilde{y} = a^{-m}(-x\sin\theta + y\cos\theta) \quad (11)$$

Where $a > 1$ and $\theta$ =nπ/N.

And,

$$a = (U_h/U_l)^{\frac{1}{M-1}} \quad (12)$$

$$W_{m,n} = a^m U_l \quad (13)$$

$$\sigma_{x,m,n} = \frac{(a+1)\sqrt{2\ln 2}}{2\pi a^m (a-1) U_l} \quad (14)$$

$$\sigma_{y,m,n} = \frac{1}{2\pi\tan\left(\frac{\pi}{2N}\right)\sqrt{\frac{U_h^2}{2\ln 2} - \left(\frac{1}{2\pi\sigma_{x,m,n}}\right)^2}} \quad (15)$$

*Wavelet Moments*

In the wake of applying Gabor filters on the picture with distinctive orientation at diverse scale, we acquire following array [5]:

$$E(m,n) = \sum_x \sum_y |G_{mn}(x,y)| \quad (16)$$

Where, $m = 0,1,\ldots,M-1;\ n = 0,1,\ldots,N-1$

It is assumed that we are interested in images or regions that have homogenous texture, therefore the mean and standard deviation are expressed as [5]:

$$\mu_{mn} = \frac{E(m,n)}{P \times Q} \quad (17)$$

$$\sigma_{mn} = \frac{\sqrt{\sum_x \sum_y (|G_{mn}(x,y)| - \mu_{mn})^2}}{P \times Q} \quad (18)$$

A feature vector $f_g$ (texture representation) is created using $\mu_{mn}$ and $\sigma_{mn}$ as the feature components [5, 6]:

$$f_g = (\mu_{00}, \sigma_{00}, \mu_{01}, \sigma_{01} \ldots\ldots. \mu_{45}, \sigma_{45}) \quad (19)$$

**Shape Features Extraction**

Shape is an imperative visual feature and it is one of the essential features used to portray content of a picture. Though, shape representation and portrayal is a hectic process. This is on the grounds that when a 3-D real world object is anticipated onto a 2-D picture plane, one measurement of object data is misplaced. Subsequently, the shape extracted from the picture partially shows to the anticipated item.

The shape features are extracted using the edge histogram descriptor (EHD). It represents the local edge distribution by dividing image space into 4×4 sub-images and representing the local distribution of each sub-image by a histogram. The fact that the EHD consists of the local-edge histograms only, makes it very flexible.

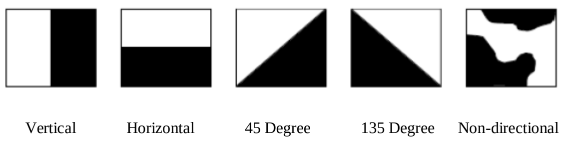


Figure 2: Five Types of Edges in the Edge Histogram Descriptor

To generate the histogram, edges in the sub-images are categorized into five types; vertical, horizontal, 45-degree diagonal, 135-degree diagonal and non-directional edges (Figure 4). Each sub-image is

IJDACR
ISSN: 2319-4863
IJDACR
International Journal Of Digital Application & Contemporary Research
International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)

further divided into non-overlapping square image blocks with particular size which depends on the image resolution. Every single block of the image is then classified into one of the five specified edge classifications or as a non-edge block. A simple method to do this classification is to treat each image-block as a 2×2 super-pixel image-block and apply appropriate oriented edge detectors to compute the corresponding edge strengths. The edge detector with maximum edge strength is then identified. If this edge strength is above a given threshold, then the corresponding edge orientation is associated with the image-block. If the maximum of the edge strengths is below the given threshold, then that block is not classified as an edge block.

*Histogram of Edge Directions*

The edge histogram is computed as follows. At first, the color image is transformed to the HSI space from which the hue channel is neglected. The resulting gradient images are next thresholded to binary images by a proper threshold value for each channel. The threshold values are manually fixed to certain levels which are the same for all images. The thresholded intensity and saturation gradient images are combined by the logical OR operation. The threshold value for the intensity gradient image was manually set to 15% of the maximum gradient value and for the saturation image to 35%. In the OR operation, the direction of the larger gradient value is chosen. Finally the 8-dimensional edge histograms are calculated by counting the edge pixels in each direction. The smoothing should make the histograms more robust to rotation. It is performed as follows:

$$H^s(i) = \frac{\sum_{l=i-k}^{i+k} H(l)}{2k+1} \qquad (20)$$

Where $H(l)$ stands for the original edge histogram and the degree of smoothing is regulated by the parameter $k$.

*Color Histogram*

Each one picture in the database is figured to acquire the color histogram, which demonstrates the extent of pixels of each one color inside the picture. The color histogram of each image is then stored in the database. When the user does the search by specifying the query image, the system registers the proportion of each color of the query image and goes through all images in the database to find those whose color histograms match those of the query most closely. The color histograms are utilized to speak to the color distribution in a picture. For the most part, the color histogram methodology checks the quantity of events of every extraordinary color on a specimen picture. Since a picture is made out of pixels and every pixel has a color, the color histogram of a picture can be processed effortlessly by going by every pixel once. By looking at the color histogram of a picture, the colors existing on the picture can be related to their relating regions as the quantity of pixels. Histogram search portrays a picture by its color distribution, or histogram. Euclidian histogram separations have been utilized to characterize the comparability of two color histogram representations.

**Classification using Random forest Classifier**

Random forests are recently proposed statistical inference tools, deriving their predictive accuracy from the nonlinear nature of their constituent decision tree members and the power of ensembles. Random forest committees provide more than just predictions; model information on data proximities can be exploited to provide random forest features. Variable importance measures show which variables are closely associated with a chosen response variable, while partial dependencies indicate the relation of important variables to said response variable.

Random Forest [7] uses decision tree as base classifier. Random Forest generates multiple decision trees; the randomization is present in two ways: (1) random sampling of data for bootstrap samples as it is done in bagging and (2) random selection of input features for generating individual base decision trees. Strength of individual decision tree classifier and correlation among base trees are key issues which decide generalization error of a Random Forest classifier [7].

Random Forest is a classifier consisting of a collection of tree-structured classifiers {h(x, $\Theta_k$) k=1, 2, ….}, where the {$\Theta_k$ } are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input x.

Random Forest generates an ensemble of decision trees. To achieve diversity among base decision trees, Breiman selected the randomization approach which works well with bagging or random subspace method. To generate each single tree in Random Forest Breiman followed following steps:

- If the quantity of records in the training set is N, then N records are examined at arbitrary however with substitution, from the first information, this is bootstrap test.
- This specimen will be the training set for developing the tree. On the off chance that there are M data variables, a number $m << M$ is chosen such that at every node, m

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

**International Journal of Digital Application & Contemporary research**
**Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)**

variables are chosen in arbitrary manner of M and the best part on these $m$ attributes are utilized to split the node.

- The estimation of $m$ is held constant within the development of forest. Each one tree is developed to the biggest degree conceivable. There is no cropping.

In this way, multiple trees are induced in the forest; the number of trees is pre-decided by the parameter $N_{tree}$. The number of variables (m) selected at each node is also referred to as $m_{try}$ or k in the literature. The depth of the tree can be controlled by a parameter node size (i.e. number of instances in the leaf node) which is usually set to one.

Once the forest is trained or built as explained above, to classify a new instance, it is run across all the trees grown in the forest. Each tree gives classification for the new instance which is recorded as a vote. The votes from all trees are combined and the class for which maximum votes are counted (majority voting) is declared as classification of the new instance. Here onwards, Random Forest means the forest of decision trees generated using this process.

In the forest building process, when bootstrap sample set is drawn by sampling with replacement for each tree, about 1/3rd of original instances are left out. This set of instances is called OOB (Out-of-bag) data. Each tree has its own OOB data set which is used for error estimation of individual tree in the forest, called as OOB error estimation. Random Forest algorithm also has in-built facility to compute variable importance and proximities [7]. The proximities are used in replacing missing values and outliers.

*Illustrating Accuracy of Random Forest:*

The Generalization error (PE*) of Random Forest is given as,

$$PE* = P_{x,y}(mg(X,Y)) < 0 \quad (21)$$

Where $mg(X,Y)$ is Margin function. The Margin function measures the extent to which the average number of votes at $(X,Y)$ for the right class exceeds the average vote for any other class. Here $X$ is the predictor vector and $Y$ is the classification.

## IV. SIMULATION RESULTS


Figure 3: Retrieved image from database

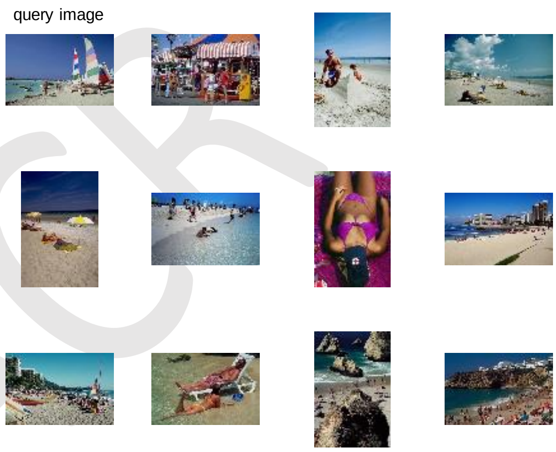


Figure 4: Retrieved image from database

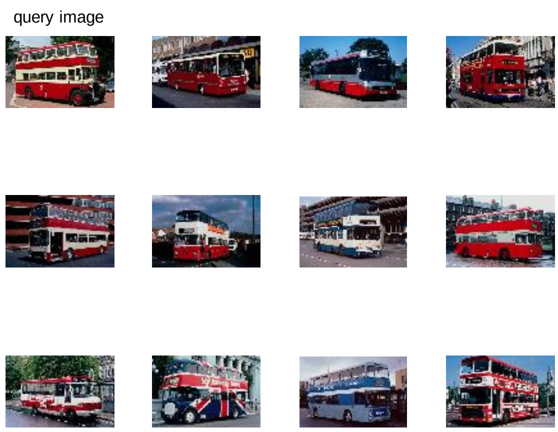


Figure 5: Retrieved image from database

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 3, Issue 7, February 2015)

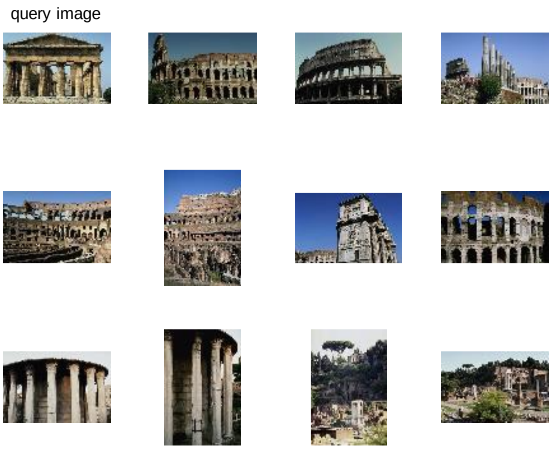


Figure 6: Retrieved image from database

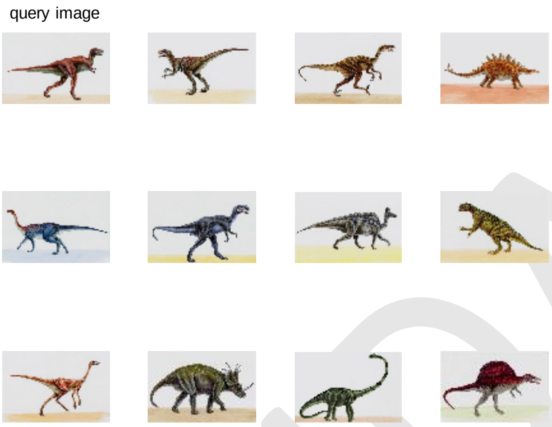


Figure 7: Retrieved image from database

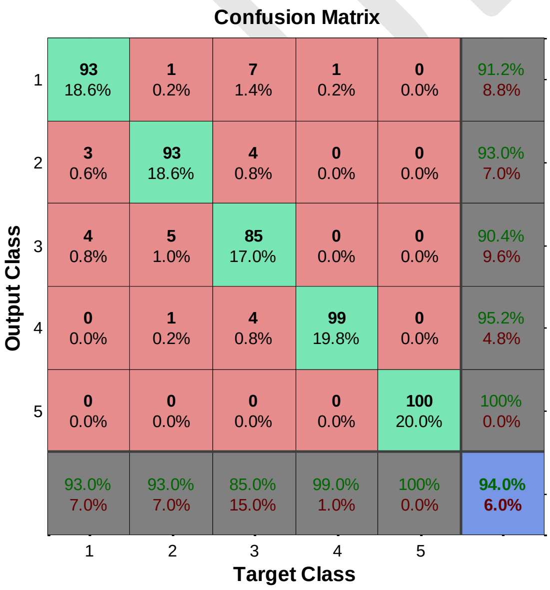


Figure 8: Confusion matrix between output class and target class

Table 1: Performance Evaluation

| Class | Category | False Negative Rate | False Positive Rate | True Positive Rate | True Negative Rate |
|---|---|---|---|---|---|
| 1 | Africa | 0.0176 | 0.0882 | 0.9118 | 0.9824 |
| 2 | Beach | 0.0175 | 0.0700 | 0.9300 | 0.9825 |
| 3 | Transportation | 0.0369 | 0.0957 | 0.9043 | 0.9631 |
| 4 | Architecture | 0.0025 | 0.0481 | 0.9519 | 0.9975 |
| 5 | Dinosaur | 0 | 0 | 1.0000 | 1.0000 |

## V. CONCLUSION

In this paper, we present a novel approach for Content Based Image Retrieval by combining the color, texture and shape features. Similarity between the images is ascertained by means of Euclidean distance. The experimental result shows that the proposed method outperforms better results in terms of False Negative Rate, False Positive Rate, True Positive Rate and True Negative Rate.

## REFERENCES

[1] Linda G. Shapiro, and George C. Stockman, Computer Vision, Prentice Hall, 2001.
[2] Barbeau Jerome, Vignes-Lebbe Regine, and Stamon Georges, "A Signature based on Delaunay Graph and Co-occurrence Matrix," Laboratoire Informatique et Systematique, Universiyt of Paris, Paris, France, July 2002.
[3] Sharmin Siddique, "A Wavelet Based Technique for Analysis and Classification of Texture Images," Carleton University, Ottawa, Canada, Proj. Rep. 70.593, April 2002.
[4] Thomas Seidl and Hans-Peter Kriegel, "Efficient User-Adaptable Similarity Search in Large Multimedia Databases," in Proceedings of the 23rd International Conference on Very Large Data Bases VLDB'97, Athens, Greece, August 1997.
[5] S. Mangijao Singh, K. Hemachandran, "Content-Based Image Retrieval using Color Moment and Gabor Texture Feature", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 1, September 2012.
[6] B. S. Manjunath and W. Y. Ma. "Texture features for browsing and retrieval of large image data", IEEE Transactions on Pattern Analysis and Machine Intelligence, (Special Issue on Digital Libraries), Vol. 18 (8), pp. 837-842, August 1996.
[7] Brieman L, Random Forests Statistics, Department University of California Berkeley, 45, 5-32, 2001.