# 1. Abstract

Content based image retrieval (CBIR), is a widely discussed topic. It is a technique to perform retrieval of the images from a large database which are similar to an image given as query. CBIR has its application in domains such as medical report images, weather forecasting, historical research etc. In this context, content refers to the information present in images which can be either texture, shape or color. Now to deal with the risk of privacy leakage of images, sensitive images, need to be encrypted before being outsourced, which causes the CBIR schemes in non-encrypted domain to become unusable. Here, we propose a scheme that supports CBIR over the encrypted images without revealing sensitive information. Firstly, both color and texture features viz. color moment (CM) on color images and local binary pattern (LBP) on grayscale images are extracted to be combined to form a single feature vector. These feature vectors now represent the corresponding images. Next, the feature vectors are protected by using a homomorphic form of encryption before being stored. When a query image is received, its features are also extracted and goes through the same homomorphic form of encryption and then compared directly with the stored encrypted feature vectors set. I hope show through analysis and experiments that the proposed scheme has both retrieval accuracy and efficiency, while preserving image privacy. The similarity matching is currently being performed by Euclidian distance on non-encrypted images which I hope to replace with hamming distance(*modified version if needed*) for encrypted images in near future.

# 2. Introduction

As the volume of digital images obtained through different type of sources has grown tremendously. The demand of efficient image retrieval have also increased. Due to increase in volume though, the image storage is typically outsourced. Outsourcing images directly to remote servers(e.g. cloud), however, increases the risk of privacy leakage when images contain sensitive information. In order to protect images against privacy leakage threats, images are usually encrypted before being outsourced. Since encryption operations disrupt the image content, it becomes a challenging task to perform CBIR over encrypted images. Therefore, it is highly desirable to devise a privacy-preserving CBIR system for cloud-based encrypted image sets.

To achieve retrieval systems two basic frameworks have made major contributions, which are content based retrieval and text based retrieval.

The text based framework, uses text descriptors to annotate the images. And are also used in database to perform information retrieval. But this approach suffers from disadvantages like, requirement of human work to annotate the images and also from annotation inaccuracy and huge time consumption. Content based image retrieval (CBIR) uses contents of image (like color, shape, texture) to annotate and perform retrieval. Basic function of CBIR is to extract the contents or features of images, then to to find out their similarity.

Here, I have proposed a CBIR technique with privacy protection, using encrypted feature vectors set having a combination of color moment (color image feature) and local binary pattern (grayscale image feature). This scheme outsources encrypted image features. We first extract the said image features, then encrypt them by a homomorphic form of encryption. Then a novel method to measure the image similarity helps to avoid revealing the image similarity information in the outsourced location (e.g. cloud to a certain extent). The scheme is suitable and easy to implement in any real time applications. The rest of the paper is organized as follows: in section 3, I have discussed in briefly the working principle of the content based image retrieval. In section 4, I have illustrated the parameters used (color moment and local binary pattern) with examples. The presented scheme and the experimental results are depicted in section 5. Finally, section 6 concludes the paper with underlining future enhancements planned.

# 3. Content Based Image Retrieval - Working Principle

An image has three types of contents as texture, color, shape. Many techniques are available to extract features based on the color contents. I have used color moment (CM) and had to find out the three order color moment- 'mean', 'standard deviation' and skewness.
Various color models are available to perform image retrieval. Though RGB color space is well supported, I have used this HSB model in my CBIR scheme as RGB is not uniform. And as all three components (R, G, B) have equal importance and so those values have to be quantized with the same precision.
For texture feature out of various available methods, I have used local binary pattern (LBP).
After feature extraction, for similarity matching : I am hoping to implement a modified version of hamming distance for having better results. Currently in this CBIR scheme I have utilized Euclidian distance as a place holder.
There are two main sub-parts in this scheme: 1. Feature extraction and 2. Searching/Retrieval.
In feature extraction, the features viz. CM and LBP are extracted from images in the image_database and stored in combination in a feature vector to create a new file storage.
In searching, initially the same feature extraction algorithm is employed in the query image to extract the feature vector and subsequently the similarity is measured with the feature vector storage using some distance measure (here, currently Euclidean distance). The best results are obtained as those having minimum similarity distance.
The process flow diagram of the CBIR technique is shown below :

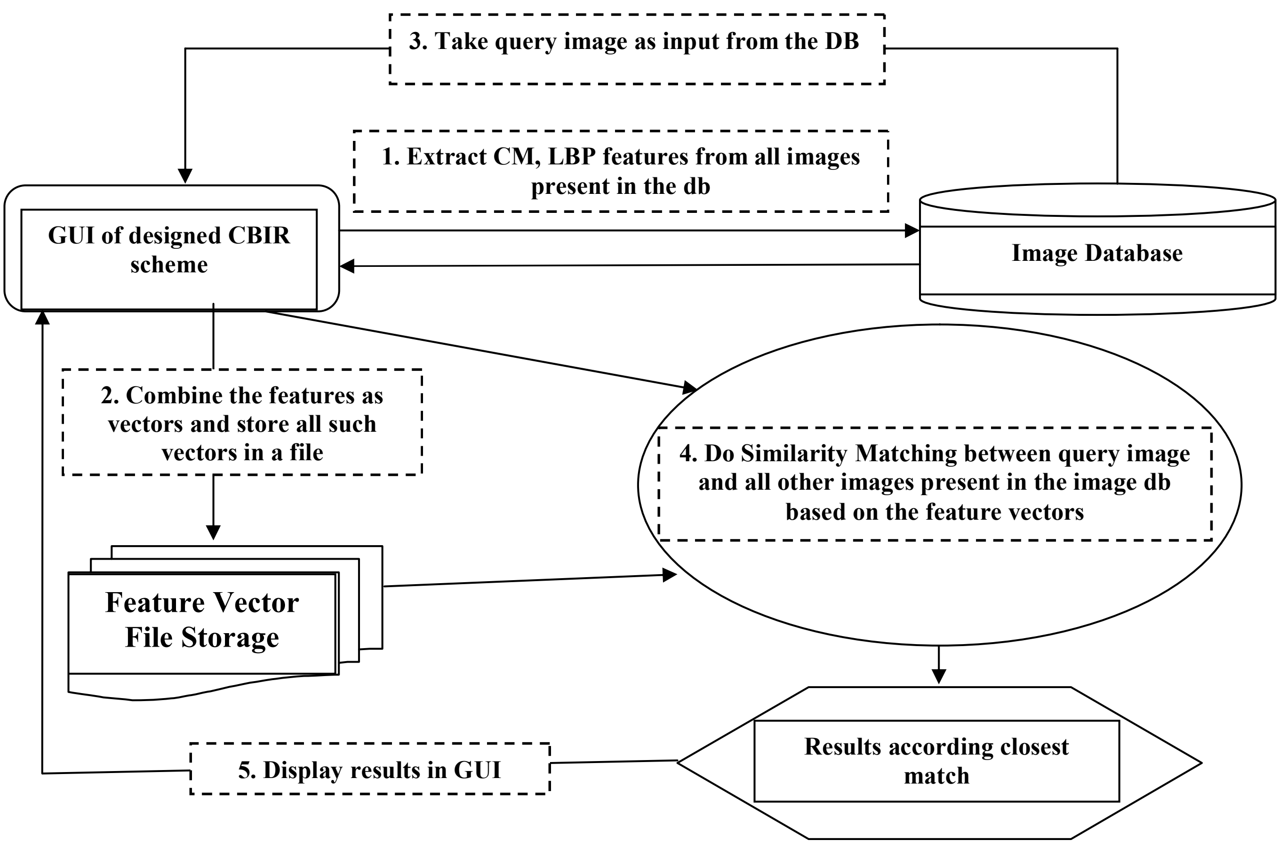


Figure : 1 – Process flow of the CBIR scheme

# 4. Background and illustrations of feature parameters used

Performance of any CBIR techniques depends on the extraction of suitable features of the images. Any such scheme should also consider that, overall computational cost should not be very high. Hence the dimension of the feature vector of the image should be selected appropriately. Here we have used two features viz. Color Moment (CM) and local binary pattern (LBP). Let us dive into their mathematical backgrounds-

## 4.1 Color Moment (CM)

Color moment is a technique that is used to extract color from image. It is used to differentiate the images based on color of image. Color moment is used to check the color similarity between images. The basis of color moments is to calculate distribution of color in an image that can be understood as a probability distribution. I have reused the method developed by Stricker and Orengo for calculating color moment, where the first, second and third order moment of each color component have been calculated.

Let the value of the $\mathbf{a}^{th}$ color channel, at the $\mathbf{b}^{th}$ image pixel is $\mathbf{P}_{ab}$, then

The first order color moments or **'mean'** is :

$$E_a = \frac{1}{N}\sum_{b=1}^{N} p_{ab}$$

The second order color moments or 'standard deviation' is:

$$\sigma_a = \sqrt{\left(\frac{1}{N}\sum_{b=1}^{N}(p_{ab} - E_a)^2\right)}$$

And, third-order color moments of 'gradient' or 'skewness' is :

$$S_a = \sqrt[3]{\left(\frac{1}{N}\sum_{b=1}^{N}(p_{ab} - E_b)^3\right)}$$

It is to be noted here, that skewness is a measure of the degree of asymmetry in the distribution and 'N' is total number of pixels in image. As I am using HSV (i.e. 3) color chanel here and each channel would have 3 moments. So the outpur of color moments calculation would be a 3X3 matrix.

### 4.2 Local Binary Pattern (LBP)

A local binary pattern was introduced by Ojala et al [5] for texture analysis. Basic LBP descriptor [4] is created by setting a threshold as the values of the 3X3 neighborhood of the pixel against the central pixel. To create LBP representation, first we have to convert the color image into grayscale image. Then we sub-block the image into 3x3 blocks. From the block, we have to find out gray level pixel values. Then we calculate LBP value. By using these LBP values we get a LBP image. The whole process is illustrated with a small example below.
Say a given 3X3 pixel neighborhood of a grayscale image is :

| | | |
|---|---|---|
| **90** | **200** | **140** |
| **180** | **172** | **100** |
| **170** | **181** | **152** |

On this the process of calculating LBP is –

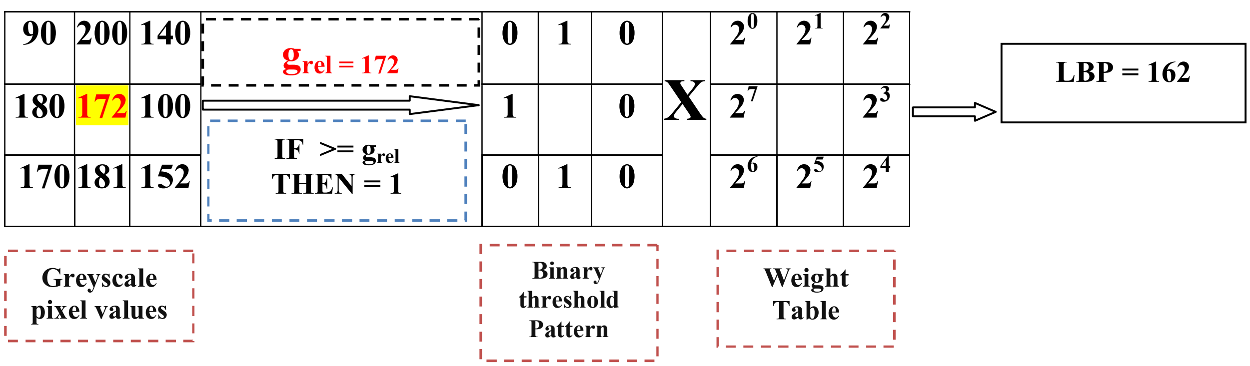


Figure : 2 – Illustration of LBP calculation

i.e. here for P = 7 and p = 0, 1, 2, 3, 4, 5, 6, 7.

$$LBP = \sum_{p=0}^{P-1} s(g_p - g_{\text{rel}}).\,2^p$$

And,

$$s(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$$

Here $g_p$ in each block is thresholded by its center pixel value $g_{\text{rel}}$ . And S(x) is the Binary threshold pattern function.

# 5. Experiment & Results

| **Algorithm 1:** | ExtractFeatures |
|---|---|

**Input** : Database of Images.
**Output** : A file storage filled with extracted feature vectors.

**Begin**:

**For** all images in the database provided -
Calculate the color moments (cm).
Convert the image into a grayscale image (*if the image is a colored one*).
Calculate the local binary pattern (lbp) in vectorized form.
Calculate the feature vector as f = [cm lbp]
**End For**

Save the feature vectors in a file storage.

**End**.

| **Algorithm 2:** | RetriveMatchingImages |
|---|---|

**Input** : Query Image.
**Output** : Retrieved images closest to the queried one.

**Begin**:

Extract feature vector for the query image.

**For** all images in the database provided -
Calculate the similarity among the image and query image.
**End For**

Pick a certain pre-fixed number of images from the image database closest to the Query image based on the used similarity measure.

Show the picked images in the retrieval UI.

**End**.

Dr. Arup Pal has provided an image database with 1000 samples. I used the same to test the proposed method. This database consist images of 10 classes. Each class has 100 images. These 10 classes are humans (Africans to be specific), beaches, monument buildings, dinosaurs, buses, elephants, flowers, horses, mountains and food.

We can see figure 3, 4 and 5 which shows the results of the algorithm when tested for closest 10, 15 and 20 matches respectively

**Case - 1 : For 10 Images :**

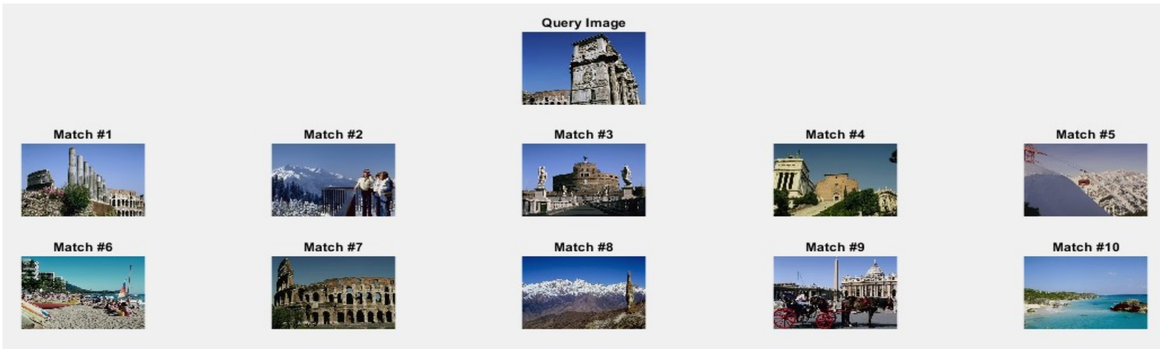


Figure : 3 – Closest 10 Matches

**Case - 2 : For 15 Images :**

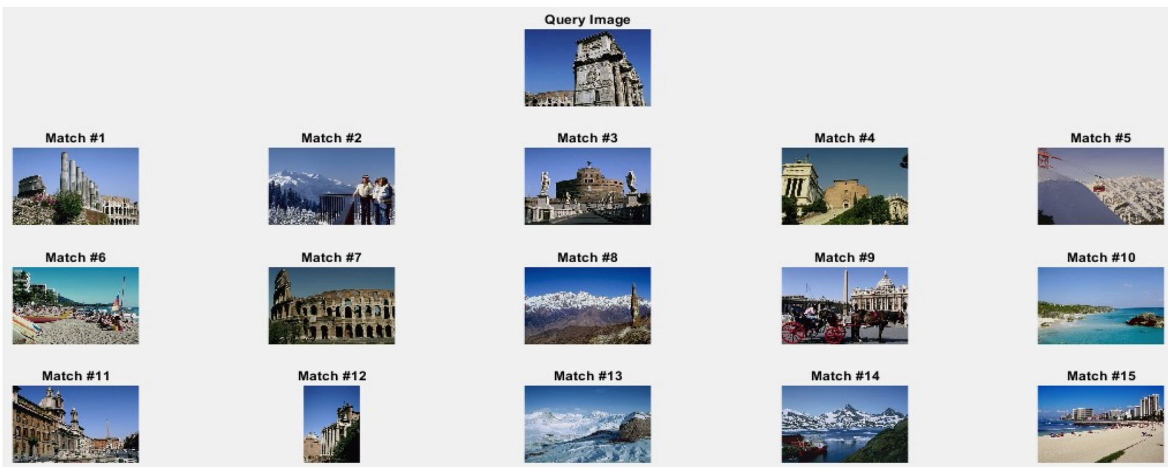


Figure : 4 – Closest 15 Matches

**Case - 3 : For 20 Images :**

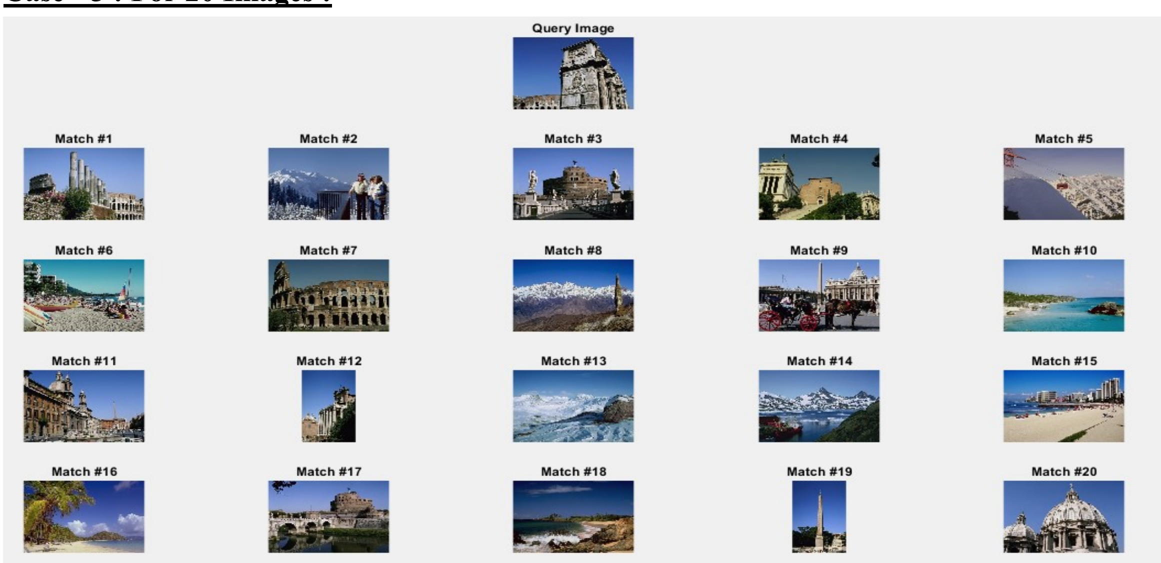


Figure : 5 – Closest 20 Matches

Now, to perform quantitive analysis, when I analysed the results, using precision and recall values defined as:

$$Precision = \frac{Number\ of\ Relevant\ images\ retrieved}{Total\ Number\ of\ images\ retrieved} \times 100$$

$$Recall = \frac{Number\ of\ Relevant\ images\ Retrieved}{Number\ of\ relevant\ images\ in\ the\ database} \times 100$$

The Calculated precision and recall values in a tabular form are as followed :-

| **Type of Images** | **Precision Values for retrieved images for** | | | **Recall Values for retrieved images** | | |
|---|---|---|---|---|---|---|
| | **10 matches** | **15 matches** | **20 matches** | **10 matches** | **15 matches** | **20 matches** |
| Humans | 0.8 | 0.87 | 18/20 | 0.08 | 0.13 | 0.18 |
| Beaches | 0.6 | 0.6 | 0.45 | 0.06 | 0.09 | 0.09 |
| Monuments | 0.5 | 0.47 | 0.5 | 0.05 | 0.07 | 0.1 |
| Buses | 0.5 | 0.47 | 0.5 | 0.05 | 0.07 | 0.1 |
| Dinosaurs | 1 | 1 | 1 | 0.1 | 0.15 | 0.2 |
| Elephants | 0.1 | 0.2 | 0.15 | 0.01 | 0.03 | 0.03 |
| Flowers | 0.8 | 0.8 | 0.85 | 0.08 | 0.12 | 0.17 |
| Horses | 0.9 | 0.9 | 0.85 | 0.09 | 0.14 | 0.17 |
| Mountains | 0.3 | 0.4 | 0.4 | 0.03 | 0.06 | 0.08 |
| Foods | 0.3 | 0.3 | 0.4 | 0.03 | 0.05 | 0.08 |

**Table -1 : Precision & Recall Values of Tested Data Sets**

# 6. Conclusion & Future scope

In this paper I have proposed and demonstrated an approach to perform content based image retrieval using a combination of extracted color and texture feature from images. As it can be seen from the quantitative analysis, the precision and recall values are not uniformly efficient accross all different types of images. Hence there are lot of scope for future works. Dr. Arup Kr. Pal has suggested some modifications, I would try to work on the them in future to improve the schemes performance :

- Rather than calculating color moments from the whole image, extract this value on a block level from each image.
- Use a modified version of hamming distance rather than currently used euclidien one to have better similarity matching.
- Make sure that the used distance measure is applicable to images that has gone through homomorphic form of encryption.
- Enhance the CBIR scheme, so that in can be used for encrypted images also that uses an homomorphic form of encryption.
- Security Analysis on the enhanced CBIR scheme.
- Performance evaluation for the encypted images.

# 7. References


[I ] Annesha Malakar, Joydeep Mukherjee, "Image Clustering using Color Moments, Histogram, Edge and K-means clustering" , International Journal of Science and Research (fJSR), Yol.2, No. 1 , pp 532-537, January 2013.

[2] Aman Chadha , Sushm it Mallik, Ravdeep Johar, "Comparative Sudy and Optimization of Feature-Extraction Techniques for Content based Image Retrieval" , intemational Journal of Computer Application s, Yo152, No .20, August 2012.

[3] Zhi-chun huang, Patrick P. K. Chan, Wing W. Y. Ng, D aniel s. Yeung " Content-based image retrieval using color moment and Gabor texture feature", Proceedings of the Ninth international Conference on Machine Learning and Cybernetics(fEEE ), Vol. 2, pp 719-724, July 2010.

[4] Caifeng Shan, "Learning local binary patterns for gender classification on real-world face images", Pattern Recognition Lel1ers, Vo1.33, No.4, pp 431-437, March 2012

[5] T. Ojala, M. Pietikinen, D. Harwood, "A comparative study of texture measures with classification based on f e atured distributi ons," Pattern Recognition, Vo1.29, N o. 1, pp. 51-59, 1996.