**CO-OCCURRENCE OF ADJACENT SPARSE LOCAL TERNARY PATTERNS TEXTURE FOR FACE IMAGE RETRIEVAL**

**Project ID:14411**

*B.Tech. Project Report*

*Submitted for fulfillment of*

*the requirements for*

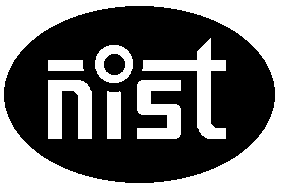
*the Degree of Bachelor of Technology*

*Under Biju Pattnaik University of Technology*

*Submitted By*

**SUPREET PRAHARAJ ROLL NO. CSE201510362**

**SUNAINA KUMARI ROLL NO. CSE201511367**



*2018- 2019*

*Under the guidance of*

**Mr. PRADEEP KUMAR JENA**

**NATIONAL INSTITUTE OF SCIENCE & TECHNOLOGY**

**Palur Hills, Berhampur, Odisha – 761008, India**

##### BONAFIED CERTIFICATE



*This is to certify that the B.Tech. Project entitled “CO-OCCURRENCE OF ADJACENT SPARSE LOCAL TERNARY PATTERNS TEXTURE FOR FACE IMAGE RETRIEVAL” is a bona fide record by* ***Supreet Praharaj*** *(Roll No. BTECH 201510362) and* ***Sunaina Kumari*** *(Roll No. BTECH 201511367) under my supervision and guidance, in partial fulfilment of the requirements for the award of Degree of Bachelor of technology from National Institute of Science and Technology under Biju Pattnaik University of Technology for the year 2018.*

#### 

#### 

#### Mr. Pradeep Kumar Jena

(Faculty advisor)

# ABSTRACT

Searching for similar face images in large databases is a time-consuming work and an efficient face image retrieval system is valuable in this situation. Content-based face image retrieval is searching for similar face images to a given query face image in a large database. The feature extraction method adopted for extracting visual features from the given face image, plays an important role in CBFIR. Texture is a prominent feature of image which can extract contents of image, hence a good texture feature extractor is necessary for content-based image retrieval process. In this project, a new texture descriptor is developed which is a combination of Local Ternary Pattern (LTP) and gray level co-occurrence matrix (GLCM). This feature descriptor which is named as CoALTP, inherits the attributes of both LTP and GLCM. First LTPs of pixels are obtained and then using GLCM in four directions, co-relations between pixel pairs are calculated as features. The PCA is used in order to reduce the dimension of the extracted features.

# ACKNOWLEDGEMENT

We would like to take this opportunity to thank all those individuals whose invaluable contribution in a direct or indirect manner has gone into the making of this project a tremendous learning experience for me.

It is our proud privilege to epitomize our deepest sense of gratitude and indebtedness to our faculty guide, **Mr. Pradeep Kumar Jena** for his valuable guidance, keen and sustained interest, intuitive ideas and persistent endeavor. His guidance and inspirations enabled us to complete our report work successfully.

We give our sincere thanks to **Mr. Debananda Kanhar, Project Coordinator**, for giving us the opportunity and motivating us to complete the project within stipulated period of time and providing a helping environment.

We acknowledge with immense pleasure the sustained interest, encouraging attitude and constant inspiration rendered by **Prof. Sangram Mudali** (Director) **Prof. (Dr.) Ajit Kumar Panda** (Dean), **Prof. (Dr.) Shom Prasad Dash** (HOD, School of Computer Science) N.I.S.T. Their continued drive for better quality in everything that happens at N.I.S.T. and selfless inspiration has always helped us to move ahead.

# TABLE OF CONTENT

[ABSTRACT 3](#_Toc4663644)

[ACKNOWLEDGEMENT 4](#_Toc4663645)

[TABLE OF CONTENT 5](#_Toc4663646)

[LIST OF FIGURES 6](#_Toc4663647)

[1. INTRODUCTION 7](#_Toc4663648)

[2. CONTENT BASED FACE IMAGE RETRIEVAL (CBFIR) 9](#_Toc4663649)

[2.1 Definition 9](#_Toc4663650)

[2.2 History of CFBIR 9](#_Toc4663651)

[2.3 Principle of CBFIR 10](#_Toc4663652)

[2.4 Applications of CBFIR 11](#_Toc4663653)

[3. LOCAL BINARY PATTERN (LBP) 13](#_Toc4663654)

[3.1 Definition 13](#_Toc4663655)

[3.2 LBP Methodology 13](#_Toc4663656)

[3.3 Properties of LBP 15](#_Toc4663657)

[4. LOCAL TERNARY PATTERN(LTP) 16](#_Toc4663658)

[4.1 Definition 16](#_Toc4663659)

[4.2 Methodology 16](#_Toc4663660)

[5. GREY LEVEL CO-OCCURRENCE MATRIX (GLCM) 17](#_Toc4663661)

[6. Principle Component Analysis (PCA) 18](#_Toc4663662)

[6.1 How does PCA work 18](#_Toc4663663)

[7. PROPOSED METHOD 19](#_Toc4663664)

[8. RESULT ANALYSIS 21](#_Toc4663665)

[9. RETRIEVAL SYSTEM 26](#_Toc4663666)

[10. PERFORMANCE EVALUATION 27](#_Toc4663667)

[11. ADVANTAGES 31](#_Toc4663668)

[12. CONCLUSION 32](#_Toc4663669)

[13. REFERENCE 33](#_Toc4663670)

# LIST OF FIGURES

[Figure 1. Content Based Face Image Retrieval System 8](#_Toc4705665)

[Figure 2. Calculation of Local Binary Pattern 15](#_Toc4705666)

[Figure 3. LBP with different radius and points 15](#_Toc4705667)

[Figure 4. LTP Calculation for a given window 16](#_Toc4705668)

[Figure 5. Calculation of GLCM with parameters θ=0o and d=1 17](#_Toc4705669)

[Figure 6. GLCM calculation with different distances and angles 17](#_Toc4705670)

[Figure 7. Flowchart of feature extraction and reduction 20](#_Toc4705671)

[Figure 8. Layer wise Input and Output shapes 20](#_Toc4705672)

[Figure 9. Sample images of ORL Face dataset 21](#_Toc4705673)

[Figure 10. Gray scale image and four LTP images 22](#_Toc4705674)

[Figure 11. Sixteen Gray Level Co-occurrence Matrices 22](#_Toc4705675)

[Figure 12. CoALTP matrix in excel format 23](#_Toc4705676)

[Figure 13. City Block Distance Matrix in excel format 24](#_Toc4705677)

[Figure 14. Euclidean Distance Matrix in excel format 24](#_Toc4705678)

[Figure 15. City Block Distance matrix in excel format 25](#_Toc4705679)

[Figure 16. Face Retrieval System of a query face 26](#_Toc4705680)

[Figure 17. Average Precision of Different Distance Metrics 28](#_Toc4705681)

[Figure 18. Average Recall Rate of Different Distance Metrics 28](#_Toc4705682)

[Figure 19. Class Wise Precision 29](#_Toc4705683)

[Figure 20. Class Wise Recall Rate 29](#_Toc4705684)

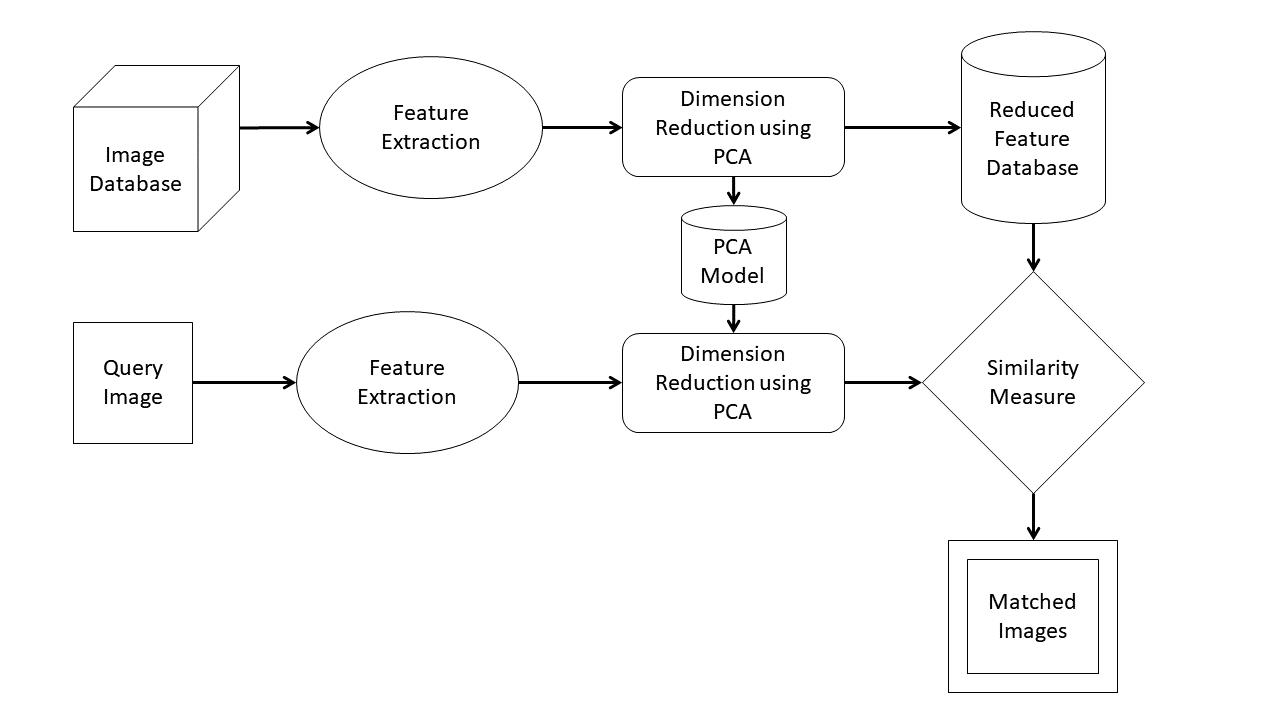
[Figure 21. AP and ARR for different number of retrieved images 29](#_Toc4705685)

[Figure 22. Comparison between different Features 30](#_Toc4705686)

# INTRODUCTION

Content based face image retrieval (CBFIR) is the application of Computer vision techniques to the face image retrieval problem and face recognition problem, that is, the problem of searching for digital face images in large databases. The Content Based Face Image Retrieval tries to solve this problem as it provides the means to index, search and retrieve specific face images. Content Based Face Image Retrieval is a task of searching face images of a person from a database and retrieval of face images, which are seems to be visually similar to a given example or query face image. Content-based Face image retrieval uses the visual contents of an image such as color, shape, texture, and spatial layout to represent and index the image. In typical content-based face image retrieval systems, the visual contents of the images in the database are extracted and described by multi-dimensional feature vectors. These feature vectors can be computed by different methods available to the users. The CBIR system consists of following components:

* **Query Face image:** It is the image to be search in the image database whether the same image is present or not or how many are similar kind images are existing or not.
* **Face Image Database:** It consists of n number of images depends on the user choice.
* **Feature extraction:** It extracts visual information from the image and saves them as features vectors in a features database. The feature extraction finds the image description in the form of feature value (or a set of value called a feature vector) for each pixel. These feature vectors are used to compare the query with the other images and retrieval.
* **Dimension reduction:** The extracted feature vector contains a huge number of variables. The feature vector is dimensionally reduced in order to comparisons faster.
* **Face Image matching:** The information about each image is stored its feature vectors for computation process and these feature vectors are matched with the feature vectors of query image which helps in measuring the similarity.
* **Resultant Retrieved face images:** It searches the previously maintained information to find the matched images from database. The output will be the similar images having same or very closest features as that of the query image.



### Figure 1. Content Based Face Image Retrieval System

# CONTENT BASED FACE IMAGE RETRIEVAL (CBFIR)

## **Definition**

CBFIR or Content Based Face Image Retrieval is the retrieval of face images based on visual features such as color, texture and shape. Reasons for its development are that in many large face 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 associating it with a keyword or number, to associating it with a categorized description, have become obsolete. This is not CBFIR. In CBFIR, each face image that is stored in the database has its features extracted and compared to the features of the query face image. It involves two steps:

* **Feature Extraction:** The first step in the process is extracting image features to a distinguishable extent.
* **Matching:** The second step involves matching these features to yield a result that is visually similar.

## **History of CFBIR**

Many would say that the father of facial recognition was Woodrow Wilson Bledsoe. Working in the 1960s, Bledsoe developed a system that could classify photos of faces by hand using what’s known as a RAND tablet, a device that people could use to input horizontal and vertical coordinates on a grid using a stylus that emitted electromagnetic pulses. The system could be used to manually record the coordinate locations of various facial features including the eyes, nose, hairline and mouth.

These metrics could then be inserted in a database. Then, when the system was given a new photograph of an individual, it was able to retrieve the image from the database that most closely resembled that individual. At the time, face recognition was unfortunately limited severely by the technology of the era and computer processing power. However, it was an important first step in proving that face recognition was a viable biometric.

In the 1970s, Goldstein, Harmon, and Lesk were able to add increased accuracy to a manual facial recognition system. They used 21 specific subjective markers including lip thickness and hair color in order to identify faces automatically. As with Bledsoe’s system, the actual biometrics had to still be manually computed.

In 1988, Sirovich and Kirby began applying linear algebra to the problem of facial recognition. What became known as the Eigenface approach started as a search for a low-dimensional representation of facial images. Sirovich and Kriby were able to show that feature analysis on a collection of facial images could form a set of basic features. They were also able to show that less than one hundred values were required in order to accurately code a normalized face image.

In 1991, Turk and Pentland expanded upon the Eigenface approach by discovering how to detect faces within images. This led to the first instances of automatic face recognition. Their approach was constrained by technological and environmental factors, but it was a significant breakthrough in proving the feasibility of automatic facial recognition.

Beginning in 2010, Facebook began implementing facial recognition functionality that helped identify people whose faces may be featured in the photos that Facebook users update daily. While the feature was instantly controversial with the news media, sparking a slew of privacy-related articles, Facebook users at large did not seem to mind. Having no apparent negative impact on the website’s usage or popularity, more than 350 million photos are uploaded and tagged using face recognition each day.

Apple released the iPhone X in 2017, advertising face recognition as one of its primary new features. The face recognition system in the phone is used for device security. The new model of iPhone sold out almost instantly, proving that consumers now accept facial recognition as the new gold standard for security.

## **Principle of CBFIR**

A typical CBFIR system as shown in Figure 2 automatically extract visual attributes (color, shape, texture and spatial information) of each face image in the database based on its pixel values and stores in a different database within the system called feature database. The feature data for each of the visual attributes of each face image is very much smaller in size compared to the face image data. Thus, the feature database contains an abstraction (compact form) of the face images in the face image database; each face image is represented by a compact representation of its contents (color, texture, shape and spatial information) in the form of a fixed length real-valued multicomponent feature vectors or signature. The users usually formulate query face image and present to the system. The system automatically extracts the visual attributes of the query image in the same mode as it does for each database image, and then identifies faces in images in the database whose feature vectors match those of the query face image, and sorts the best similar objects according to their similarity value. During operation the system processes less compact feature vectors rather than the large size image data thus giving CBFIR its cheap, fast and efficient advantage over text-based retrieval. CBFIR system can be used in one of two ways. First, exact image matching, that is matching two images, one an example image and the other, image in image database [Face Recognition]. Second is approximate image matching, which is finding most closely match images to a query image [Face Images Retrieval].

## **Applications of CBFIR**

It is an enabling technology for many applications including automatic face annotation, crime investigation, etc.

**Payments:** In 2016, MasterCard launched a new selfie pay app called MasterCard Identity Check. Customers open the app to confirm a payment using their camera, and that’s that. Facial recognition is already used in store and at ATMs, but the next step is to do the same for online payments. Chinese ecommerce firm Alibaba and affiliate payment software Alipay are planning to apply the software to purchases made over the Internet.

**Security and Access:** As well as verifying a payment, facial biometrics can be integrated with physical devices and objects. Instead of using passcodes, mobile phones and other consumer electronics will be accessed via owners’ facial features. Apple, Samsung and Xiaomi Corp. have all installed Face Retrieval Technology in their phones. This is only a small-scale example, though. In future, it looks like consumers will be able to get into their cars, houses, and other secure physical locations simply by looking at them. Jaguar is already working on walking gait ID – a potential parallel to facial recognition technology. Other corporations are likely to take advantage of this, too.

**Criminal Identification:** If Face Retrieval Technology can be used to keep unauthorized people out of facilities, surely it can be used to help put them firmly inside them. This is exactly what the US Federal Bureau of Investigation is attempting to do by using a machine learning algorithm to identify suspects from their driver’s licenses. The FBI currently have a database which includes half of the national population’s faces. This is as useful as it is creepy, giving law enforcers another way of tracking criminal across the country. AI equipped cameras have also been trailed in the UK to identify those smuggling contraband into prisons.

# LOCAL BINARY PATTERN (LBP)

## **Definition**

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixel of an image by thresholding the neighborhood of each pixel and considers the result as a binary number.

The Local Binary Pattern (LBP) operator is an operator that describes the surroundings of a pixel by generating a bit code from the binary derivatives of a pixel. The operator is usually applied to grey scale images and derivative of the intensities. The LBP operator takes 3x3 surrounding of a pixel and

1. Generates a binary 1 if the neighbour is greater than or equal to the centre.

2. Generates a binary 0 if the neighbour is less than the centre.

The eight neighbours of the centre can then be represented by an 8-bit number.

## **LBP Methodology**

Local binary pattern is a computationally efficient texture descriptor, transforming an image under examination into a compact image of integer labels describing at least one essential characteristic within the small-scale appearances of the image. These locally generated LBP labels are encoded into binary codes and processed to achieve a unique representation of the image invariant to changes in illumination intensities. These LBP labels are then extracted and organized into a histogram that is used for further image analysis. There are many ways to generate this histogram; one way would be to concatenate the local histograms from many small regions divided equally in the image.

A key advantage of the LBP histogram is its normalization for translation. Rotations of a textured input image can cause the LBP patterns to translate into a different location and to rotate about their origin; computing the histogram of the LBP labels normalizes the translation.

The original version of the local binary pattern operator works in a 3x3 pixel window of an image. The neighboring pixels in this window are thresholder by the center pixel intensity value, multiplied by powers of two and then summed to obtain the label for that center pixel. As the 3x3 neighborhood consists of 8 pixels, there is a total of 28 = 256 different labels that can be obtained.

Consider an image I (x, y) and let gp denote the gray value of a sampling point with coordinates xp, yp in an evenly spaced circular neighborhood of P sampling points and radius R around point xc, yc:

Assume that the local texture T of the image I (x, y) is characterized by the joint distribution of gray values of P+1 (P>0) pixels. Moreover, let gc denote the gray level of the local texture neighborhood center pixel xc, yc) i.e. gc = I (xc, yc):

Without loss of information, the center pixel value can be subtracted from the neighborhood:

The signs of the differences are considered:

where s(z) is the thresholding step functions:

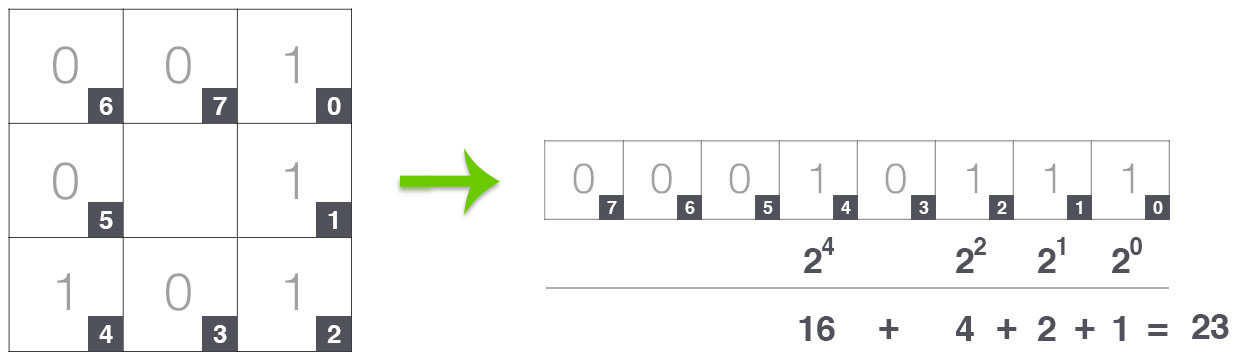
Next by summing the threshold differences, weighted by powers of two, the LBPP,R operator is defined:

LBPP,R(xc, yc) =

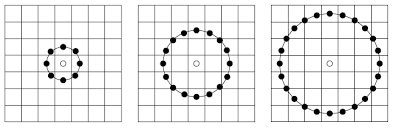
## **Properties of LBP**

LBP owns following important properties:

1. It is simple and efficient local descriptor for describing textures.
2. It encodes the relationship between the grey value of centre pixel and surrounding neighbouring pixels into 0 and 1.
3. It is helpful in extracting local information of an image.
4. As a local feature, when it is combined with global feature acts as a powerful feature vector.

****

### Figure 2. Calculation of Local Binary Pattern



### Figure 3. LBP with different radius and points

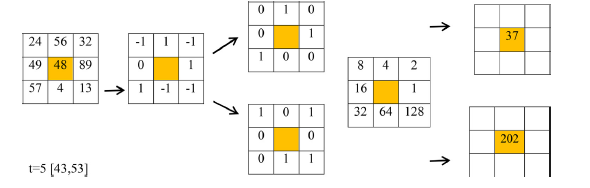
# LOCAL TERNARY PATTERN(LTP)

## **Definition**

It is an extended version of the LBP to the three valued code in which gray values in the range of Ic-th and Ic+th are quantized to 0, gray values above Ic+th to 1 and those below Ic-th to -1. LTP is calculated according to below equation for each pixel of image.

## **Methodology**

In LTP first, we take a 3x3 window of the image. From that 3x3 window we take the centre value. Then subtract and add the threshold value to the centres pixel to get two value lc-th and lc+th. Then we create two matrices: Upper LTP and Lower LTP. The Upper LTP can be calculated as placing 1 where the pixel value in window is greater than lc+th. The Lower LTP can be calculated as placing 1 where the pixel value in window is lesser than lc-th. After calculation of the two matrices we multiply both by the scale and take the sum. The resultant sum will be the value of the correspond pixel in LTP matrices.

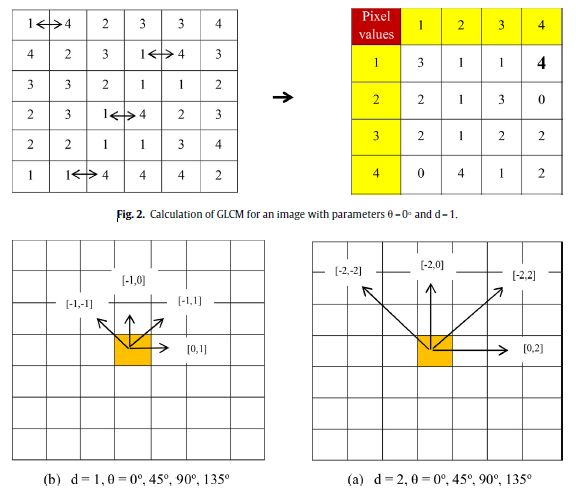


### Figure 4. LTP Calculation for a given window

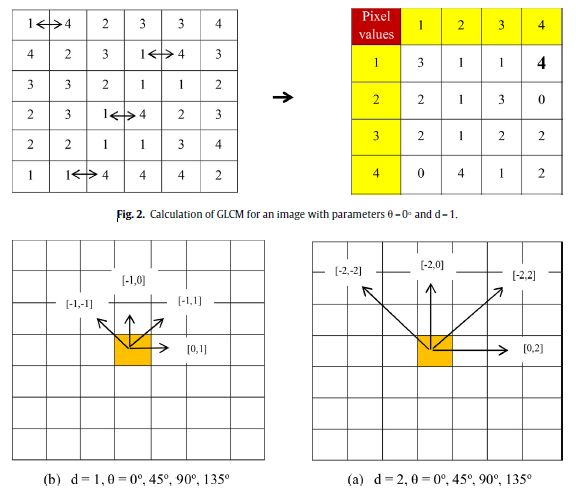
# GREY LEVEL CO-OCCURRENCE MATRIX (GLCM)

Gray level co-occurrence matrix is a matrix that is defined over an image to be distribution of co-occurring of gray-level values at a given offset. The co-occurrences are calculated with respect to a specific distance and in a given direction.

I (p, q) is the gray value of pixel at p’s row and q’s column, i and j are gray values which co-occurrence for them are calculated and 1{} is the indicator function. Δx and Δy are positional offsets in x and y directions and depend on the distance and direction which the co-relations are calculated with respect to them.



### Figure 5. Calculation of GLCM with parameters θ=0o and d=1



### Figure 6. GLCM calculation with different distances and angles

# Principle Component Analysis (PCA)

Principal Component Analysis (PCA) is a dimension-reduction tool that can be used to reduce a large set of variables to a small set that still contains most of the information in the large set. Principal component analysis (PCA) is a mathematical procedure that transforms a number of (possibly) correlated variables into a (smaller) number of uncorrelated variables called principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible.

## **How does PCA work**

1. Calculate the covariance matrix X of data points.
2. Calculate eigen vectors and corresponding eigen values.
3. Sort the eigen vectors according to their eigen values in decreasing order.
4. Choose first k eigen vectors and that will be the new k dimensions.
5. Transform the original n dimensional data points into k dimensions.

**Variance:** It is a measure of the variability or it simply measures how spread the data set is. Mathematically, it is the average squared deviation from the mean score. We use the following formula to compute variance var(x).

**Covariance:** It is a measure of the extent to which corresponding elements from two sets of ordered data move in the same direction. Formula is shown above denoted by cov(x, y) as the covariance of x and y.

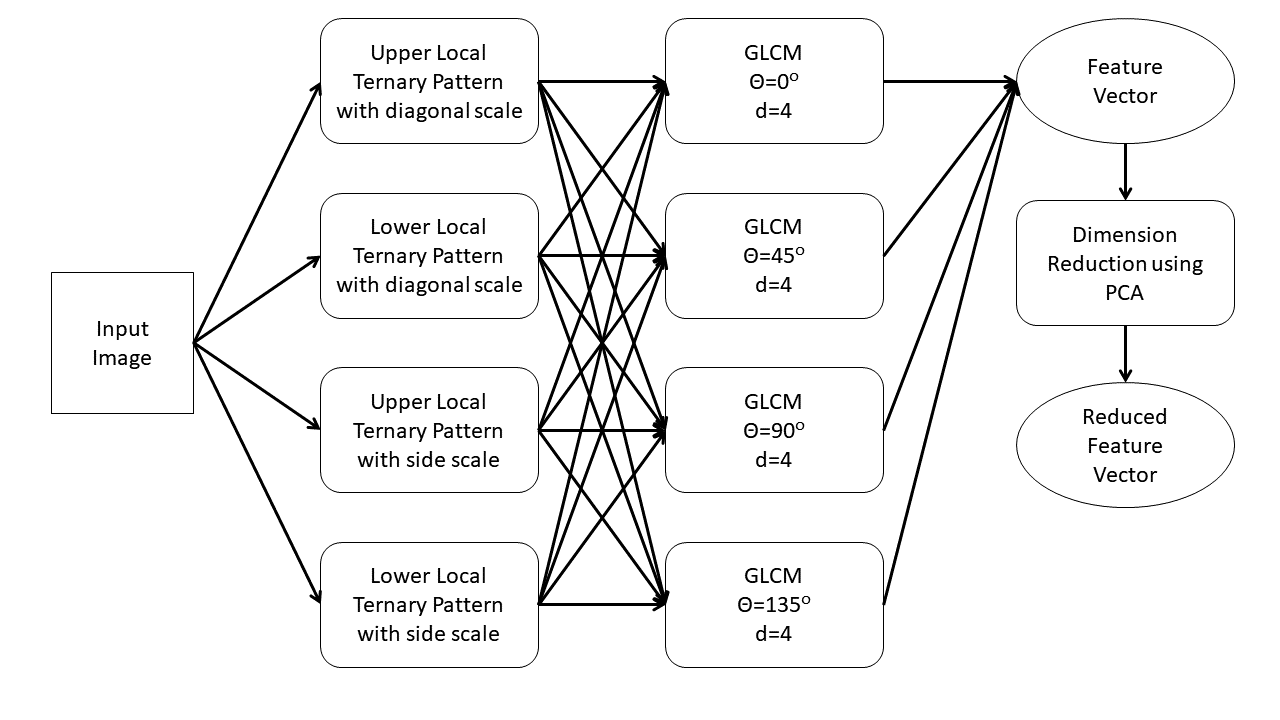
Here, xi is the value of x in ith dimension and and denote the corresponding mean values.

# PROPOSED METHOD

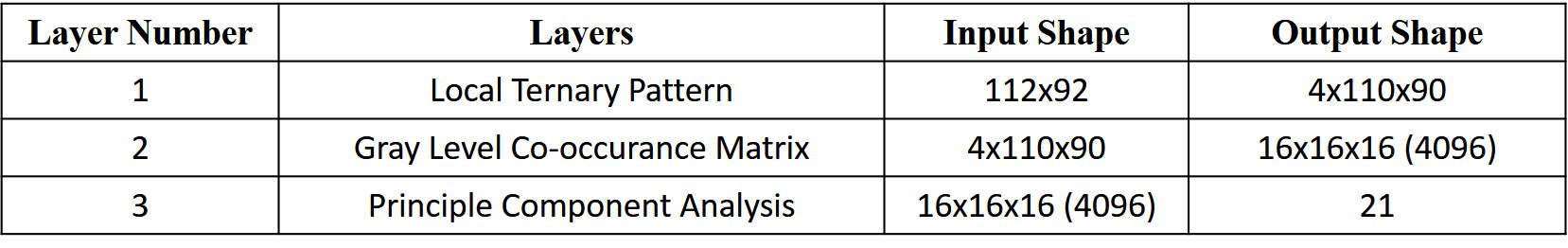
The proposed method for image retrieval consists of five steps-

1. Compute the feature vector of query face image using proposed method i.e. CoALTP technique.
2. Obtain feature vectors for all of database images using proposed method.
3. Using PCA, the feature vectors’ size is reduced.
4. Calculate the Euclidean, city block and d1 distance between the query image and each image from database.
5. Sort the obtained distances in non-decreasing order.
6. Analyze the result and compare the performance of three distances.

The LTP pattern map has the intensity values in a range of [0, 255] and in LTP two pattern maps are obtained, one corresponding to upper pattern and the other one corresponding to lower pattern, so one co-occurrence matrix for each pattern map have the size of 256 × 256 which using four co-occurrence matrices and considering upper and lower LTP pattern maps, the final feature vector have the size of (256 × 256) × 2 × 4. In order to shorten the feature vector length, sparse local ternary patterns are used in the proposed method in which instead of considering eight neighbors in LTP calculation, four diagonal neighbor pixels with a radius of 1 and four side neighbor pixels with a radius 1 are used separately. So LTPs are obtained using four diagonal neighbor pixels and four side neighbor pixels. By using this strategy, the gray value range of each pattern map (upper or lower LTPs map) becomes [0,15] and the length of final feature vector shrinks to the size of (16 × 16) × 2 × 4 x 2 = 4096. The size of feature vector makes the retrieval system slower and affect the performance of the retrieval system. This problem can be solved using dimension reduction technique. Here, we are using Principal Component Analysis that reduce the feature vector from 4096 to 21 principal components.



### Figure 7. Flowchart of feature extraction and reduction



### Figure 8. Layer wise Input and Output shapes

# RESULT ANALYSIS

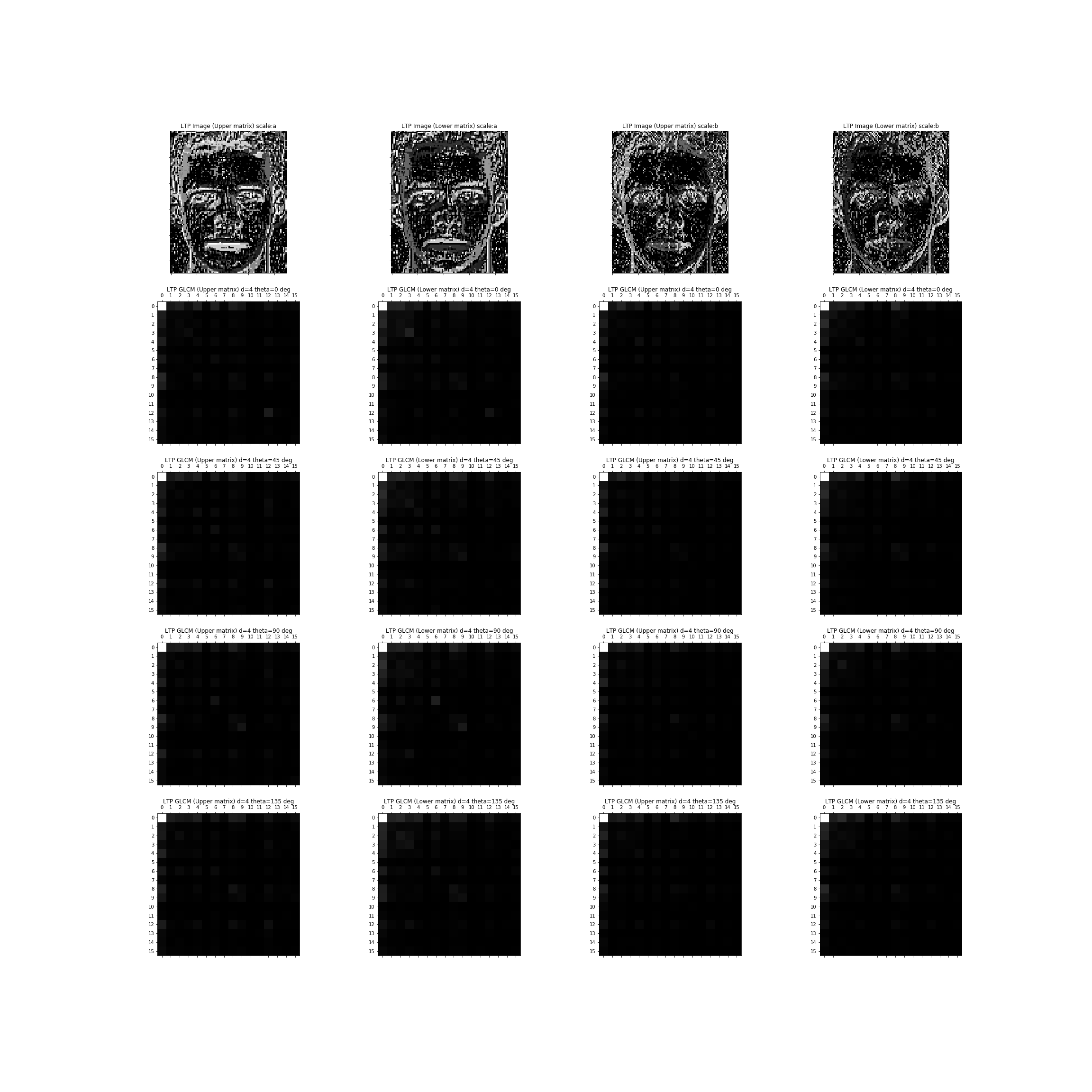
We have taken ORL FACE Dataset from AT&T for experimentation. It has 400 images of 40 persons each of having 10 images. The images of each person are stored in separate folder named as S01, S02, …, S40. Inside each folder the 10 images are named as 0.pgm, 1.pgm, …, 9.pgm.

The first step is to extract images from this dataset.

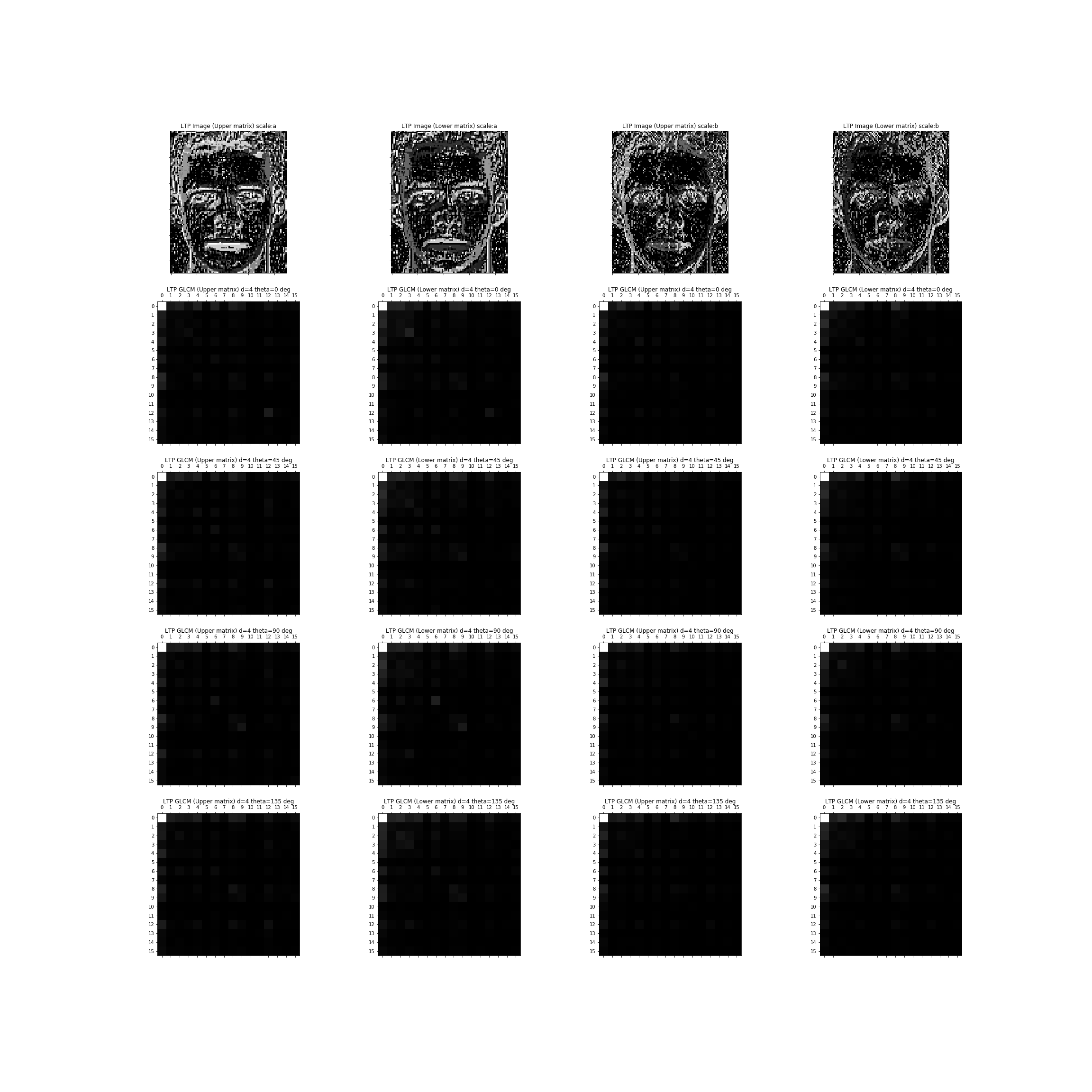
### Figure 9. Sample images of ORL Face dataset

Each of the extracted images is in the form of gray scale image. So that it becomes easier to analyze.



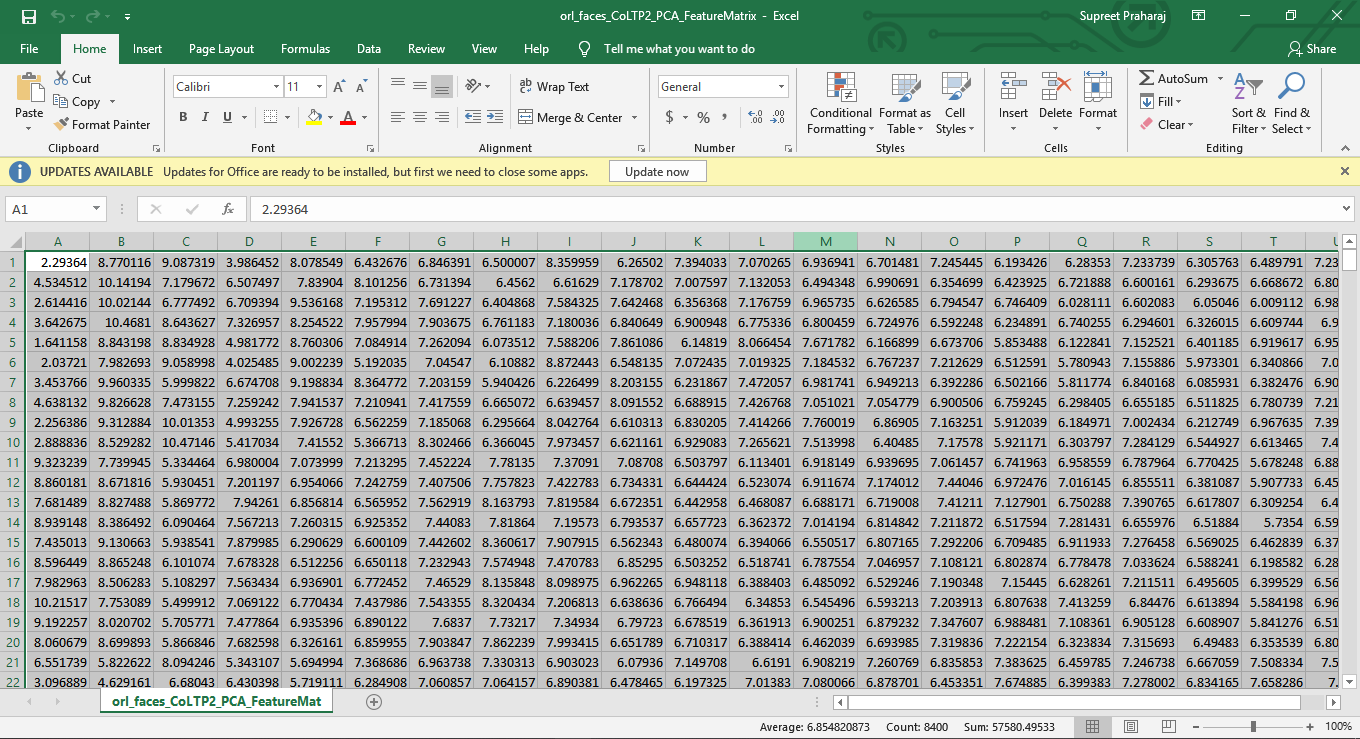


### Figure 10. Gray scale image and four LTP images



### Figure 11. Sixteen Gray Level Co-occurrence Matrices

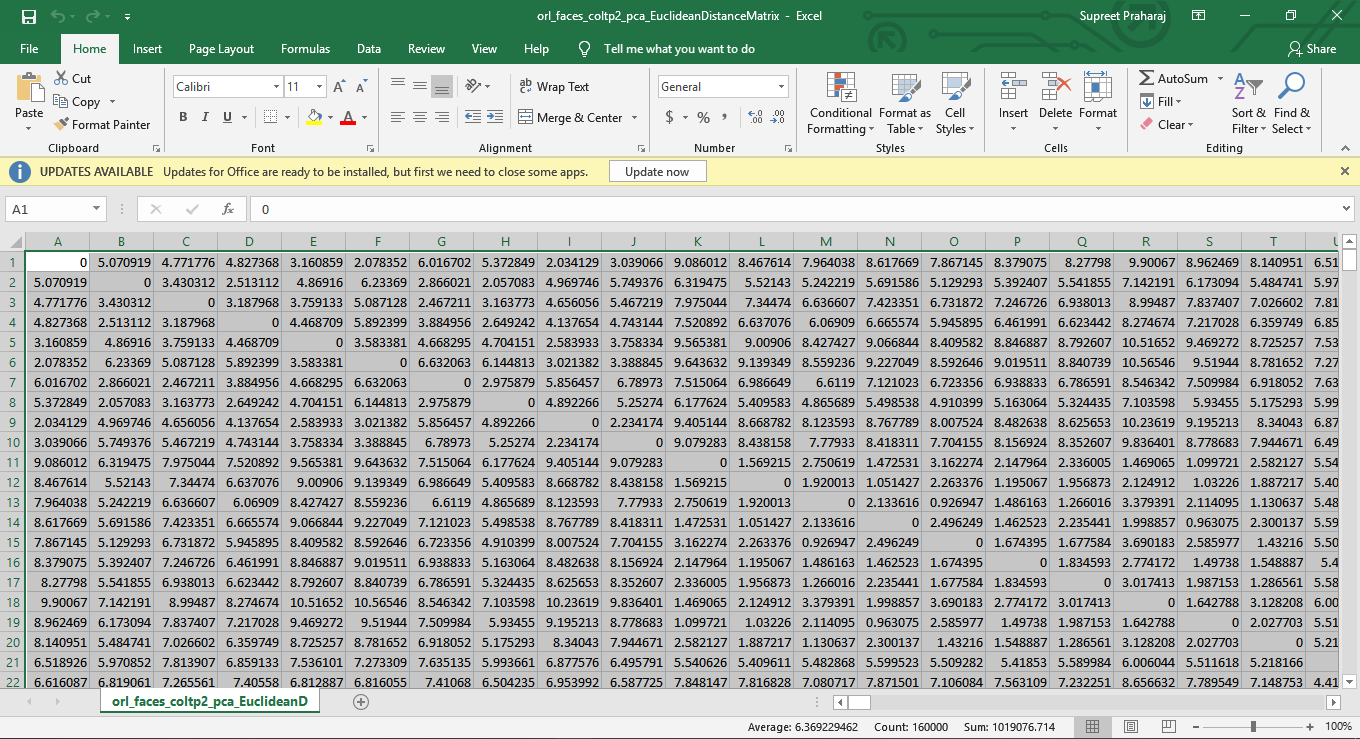
Then we assume the threshold to be 5 and find out the CoALTP using the formulae. Four different feature vectors are found and concatenated them to make one vector of length 1\*4096. Similarly, we find the feature vector of all the 400 images to create the feature matrix. Then we reduce the dimension of the feature matrix using PCA to size 400 x 21. In PCA we first create a model by fitting the CoALTP feature matrix to the PCA model. Then we transform the CoALTP feature matrix to reduced feature matrix.



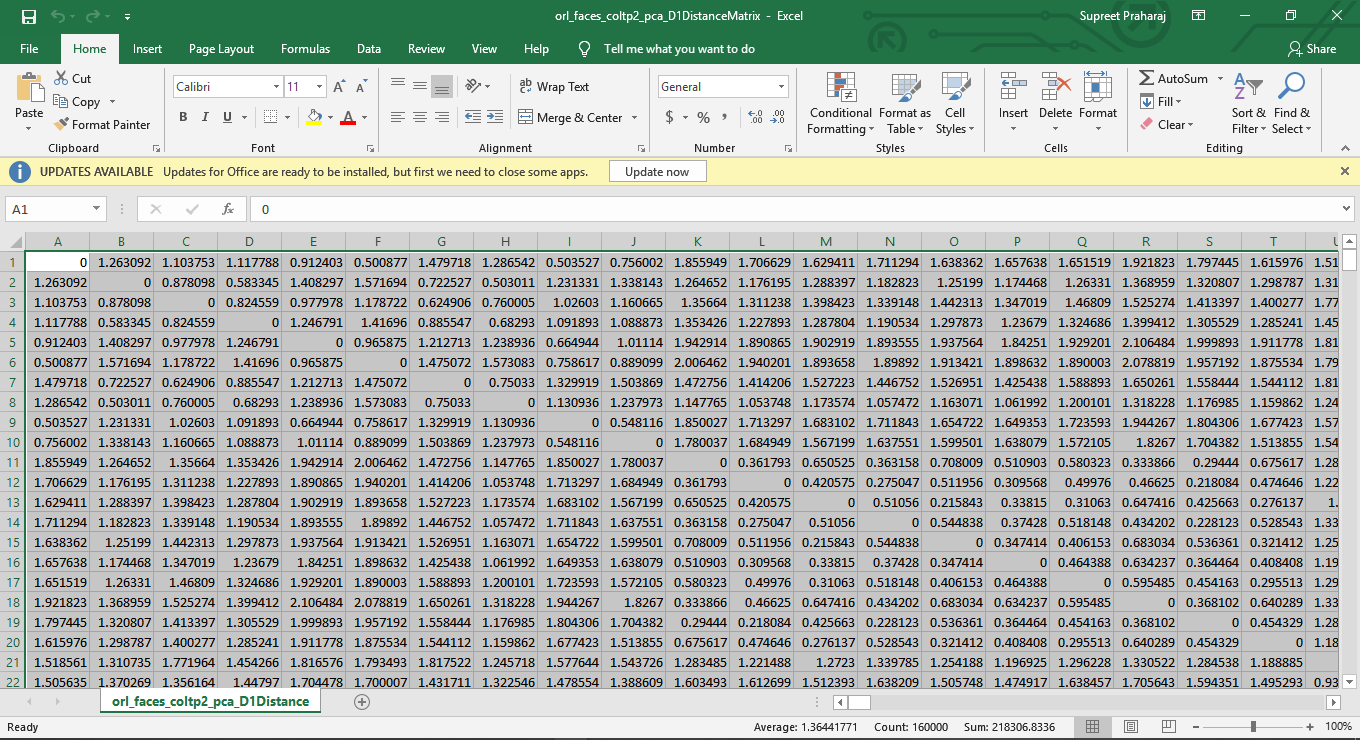
### Figure 12. CoALTP matrix in excel format

Based upon the distance of two pixels, we can find out whether the images are similar or not. Here we have calculated the City block, Euclidean distance and D1 distance using the formulae:

### Figure 13. City Block Distance Matrix in excel format



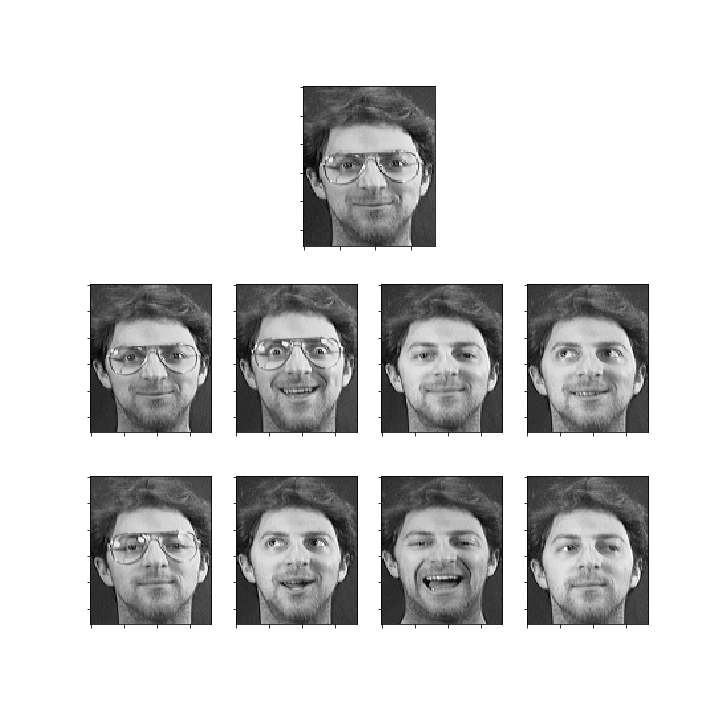
### Figure 14. Euclidean Distance Matrix in excel format



### Figure 15. City Block Distance matrix in excel format

# RETRIEVAL SYSTEM

The retrieval system takes the query image and the feature database. It calculates four LTP images followed by sixteen GLCM which produce a query feature vector of size 1x4096. Then using the PCA model that computed previously we reduce the vector to 21 query principal components. Finally, we compare the principal components with the feature data base. Here we demonstrate best 8 retrieved face images for a query image in figure 17. The image is present in S06 in the image database.



### Figure 16. Face Retrieval System of a query face

# PERFORMANCE EVALUATION

Performance of the proposed method has been evaluated in terms of precision and recall. Precision is defined as the ratio of total number of relevant facial images retrieved to the total number of facial images retrieved. Mathematically, precision can be formulated as:

where denotes total number of relevant facial images retrieved and denotes total number of facial images retrieved.

Recall is defined as the ratio of total number of relevant facial images retrieved to the total number of relevant facial images in the database.

Mathematically, recall can be formulated as

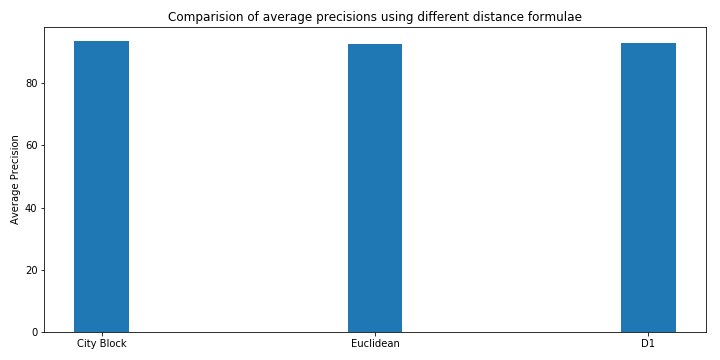
where denotes total number of relevant facial images retrieved and denotes total number of relevant facial images in the database.

For three different distances we calculate the average recall rate for 5 retrieve images are:

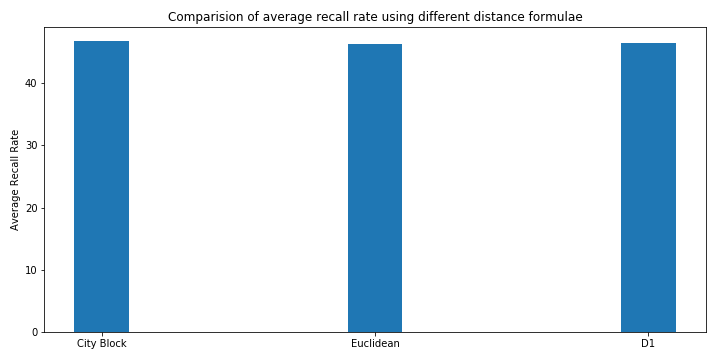
1. City Block Average Recall Rate: 46.650%
2. Euclidean Average Recall Rate: 46.225%
3. D1 Average Recall Rate : 46.450%

For three different distances we calculate the average precision for 5 retrieve images are:

1. City Block Average Precision : 93.999%
2. Euclidean Average Precision : 92.450%
3. D1 Average Precision : 92.900%

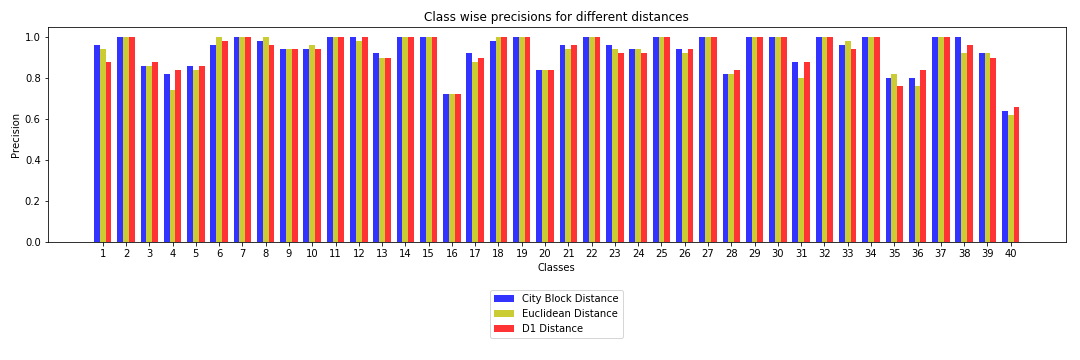


### Figure 17. Average Precision of Different Distance Metrics

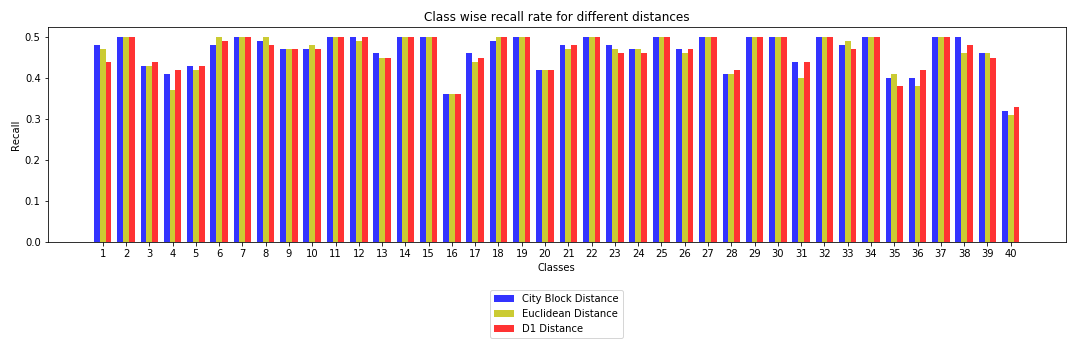


### Figure 18. Average Recall Rate of Different Distance Metrics

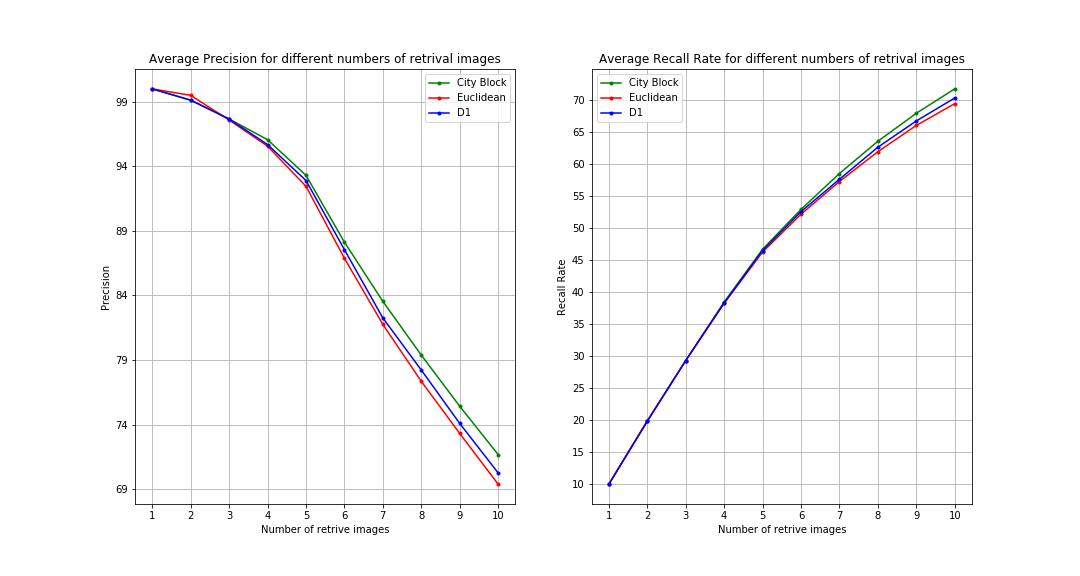
We also compute the class wise precision and recall rate for all the classes (persons) present in the database in Figure 19 and Figure 20. For each class we compare with different distance measurement formulae. The class wise precision and recall rate is computed for 30 retrieve images.



### Figure 19. Class Wise Precision



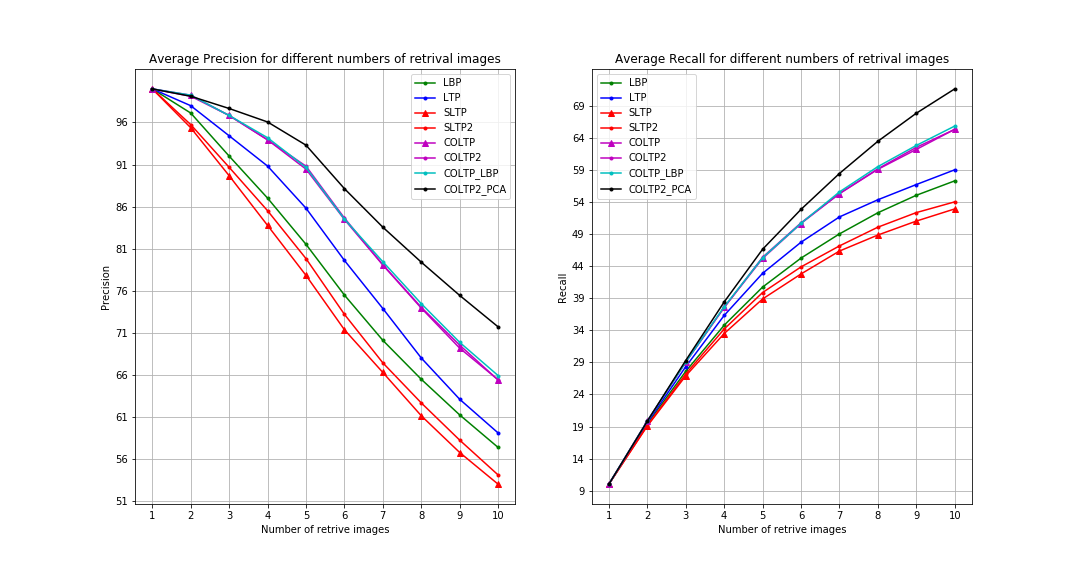
### Figure 20. Class Wise Recall Rate



### Figure 21. AP and ARR for different number of retrieved images

We find the average precision and average recall rate for various number of retrieve images range from 10 to 100 in an interval of 10: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]. The result is show in the plot in Figure 21. The average precision is not much differing for different distances but the average recall rate is higher for city block distance for every value of retrieve number.

In Figure 22, we compare the average precision and average recall rate for various feature descriptors with our proposed method. For all the values of number of retrieved images the average precision and average recall rate is best in case of CoALTP with PCA.



### Figure 22. Comparison between different Features

# ADVANTAGES

The content-based face image retrieval using CoALTP has the following advantages:

1. The combination of features help in extracting finer details in a facial image as the CoALTP technique attempts to extract local relationship among intensity values and moment attempts to gather shape information from the image. This combination covers more details of an image and constructs efficient feature vector for retrieval as compared to single feature.

2. CoALTP provides local information in an image. CoALTP provides directional information as well. This helps in getting spatial distribution of intensity values in different directions which helps in determining shape feature from texture feature of face image.

3. By using PCA, the dimension is reduced without losing information, that helps in faster processing and comparison of query image feature vector in the feature database.

# CONCLUSION

Here, an effective facial image retrieval method is presented by implementing the CoALTP technique with PCA. Here we are using the principle of Local ternary pattern and Grey level co-occurrence matrix in order to increase the accuracy and using PCA in order to make the system faster. Since this proposed method make use of these algorithms, the results obtained are more reliable.

# REFERENCE

[1] Vahid Naghashi, University College of Nabi Akram, Tabriz, Iran, Co-occurrence of adjacent sparse local ternary patterns: A feature descriptor for texture and face image retrieval, Article history: Received 7 September 2016, Accepted 22 November 2017.

[2] S. Murala, R.P.Maheshwari, R. Balasubramanian, Local maximum edge binary patterns: a new descriptor for image retrieval and object tracking, Signal. Process. 92(6) (2012)1467-1479.

[3] S. Murala, R.P. Maheshwari, R. Balasubramanian, Local tetra patterns: a new descriptor for content-based image retrieval, IEEE Trans. Image Process.21 (5) (2012)2874-2886.

[4] X. Wang, Y. Yu, H. Yang, An effective image retrieval scheme using color, Texture Shape Features Comput. Stand. Interfaces 33 (1) (2011) 59–68.

[5] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution grey-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.