**A MINOR PROJECT MID TERM REPORT**

**ON**

**FACE DETECTION AND RECOGNITION**

**Submitted by:**

Deepa Bhandari (BCE 140336)  
Rohan Baidya (BCE 140335)  
Deepak Prasad Shah (BCE 140328)  
Punita Kushwaha (BCE 140316)

**Submitted To:**

**The Department of IT and Computer Engineering**In Partial fulfillment of requirement for the degree of  
Bachelor of Engineering in Computer

****

**COSMOS COLLEGE OF MANAGEMENT AND TECHNOLOGY**(Affiliated to Pokhara University)Tutepani, Lalitpur, Nepal

January 2018

**ACKNOWLEDGEMENT**

Initially, we want to express our gratitude to our Head of Department for Bachelors in Engineering for IT and Computer Program, Er. Bibek Ropakheti, to provide this opportunity to perform extensive research on knowledge based in real world application by means of assigning us this project. We are also obliged to our seniors who have helped paved the course for us in this project to make it more authentic.

Among the list finally, we also extend our earnest appreciation to our amicable and inspiring faculty members, staffs and friends that have helped us to develop, filter and scrutinize the application and the current project report.

Sincerely,

Deepa Bhandari (BCE 140336)  
Rohan Baidya (BCE 140335)  
Deepak Prasad Shah (BCE 140328)  
Punita Kushwaha (BCE 140316)

**ABSTRACT**

Albeit all of our sense organs are significant in different ways, we are often disposed to make use of visual and audial means more frequently to perceive, recognize and use it as a means of communication. As it is eminent that human mind is the nature’s state of the art; moreover, the concept behind running of whole body system is simple yet consists of astoundingly complex neural connection. Furthermore, all the sensory perceptions via receptor organs are transmitted in form of electrical signals to brain – where the brain; as a complex combination of logic processor, processes those input signals to generate a fruitful output by means of motor organs (i.e. Muscular movement, Speech).

Current system of Face Detection and Recognition (FDR) is intended to mimic the brain in tandem with human eye, in the way we innately perceive vision and communicate information. Keeping pace with the last 60 years of continuous research and development of advanced hardware and algorithms; using computer-based recognition system has indeed evolved to become increasingly efficient, reliable and accurate.

**TABLE OF CONTENTS**

1. ACKNOWLEDGEMENT i
2. ABSTRACT ii
3. TABLE OF CONTENTS iii
4. LIST OF FIGURES iv
5. LIST OF TABLES iv
6. ABBREVIATION iv
7. INTRODUCTION 1
   1. Background 1
   2. Objective 2
8. PROBLEM STATEMENT 3
9. SCOPE 4
10. LITERATURE REVIEW 5
    1. Computer Vision 7
    2. OpenCV 8
    3. Haar-like Feature 8
    4. Cascade Classifier 9
    5. Haar Cascade 9
    6. Block diagram 11
11. METHODOLOGY 11
12. TESTING 14
13. OUTPUT 16
14. SCHEDULE 17
15. CONCLUSION 18
16. REFERENCE 19

**LIST OF FIGURES**

1. A single 8-bit Color and Greyscale pixel 3
2. Face Recognition Process Steps
3. Haar-Feature 8
4. Block Diagram for Face Recognition
5. Class Diagram for Face Recognition
6. Output of Face Recognition (till Face Detection)
7. Face Database of SQLiteStudio under construction

**LIST OF TABLES**

1. Gantt Chart Table 15

**LIST OF ABBREVIATION**

1. Adaboost: Adaptive Boosting
2. AI: Artificial Intelligence
3. ASP: Application Specific Processor
4. BSD: Berkley Software Distribution
5. CCTV: Closed Circuit Television
6. DNN: Deep Neural Network
7. DVC: Digital Video Camera
8. GPU: Graphics Processing Unit
9. OpenCV: Open source Computer Vision
10. RGB: Red – Green – Blue
11. **INTRODUCTION**
    1. **Background**

Facial recognition is a category in biometrics that maps an individual's facial features mathematically measured using numerical code called faceprint. Faceprint is stored in database describing facial landmarks or nodal points (as named by FaceIT®). Some of those are points are as:

* Distance between the eyes
* Width of the nose
* Depth of the eye sockets
* The shape of the cheekbones
* The length of the jaw line

This is analogous to the way how humans identify one another. The face detection algorithm also processes faceprints for provided facial pattern of source image and hence return the value with data of image consisting of highest match from database.

Facial recognition alone can’t take a big leap forth. It has a forerunner that implies on first detecting the part of the image that consists of human face, which is a phase called Face Detection.

Face Detection uses mathematical computation to separate parts of face in an image (cluster). This secludes only the facial parts of an image for further processing. Moreover, Face Detection and Recognition occur in tandem and secluded from user perspective than being separate.

* 1. **Objective**

Facial Recognition has a huge implication in today’s technology and has been a matter of research since last 50 to 60 years. Lot of developments have been made in the past 20 years inclusively in hardware elements as sensors and CPU, GPU or even ASPs and algorithms to detect and recognize face with high accuracy. Coinciding with the current level of knowledge we hence, designate following ambition for choosing this project:

* To make a real time application to stream video:  
  Here, we design an application to activate and fetch the video from the web camera or DVC peripherals attached with the system and show it real time on screen.
* To add the sample image or train set and supply data:  
  The application would be able to accept the captured image and supply it to the database as a train set or sample (template) image with the data related to the subject.
* To detect and recognize the face real time using faceprint:  
  The facial image on the real time video is detected and highlighted on the display then identified using faceprint with closes match on the database.
* To load the data from database of specified faceprint:  
  The data related to the subject matching the faceprint from the database is extracted and then displayed real time on screen.

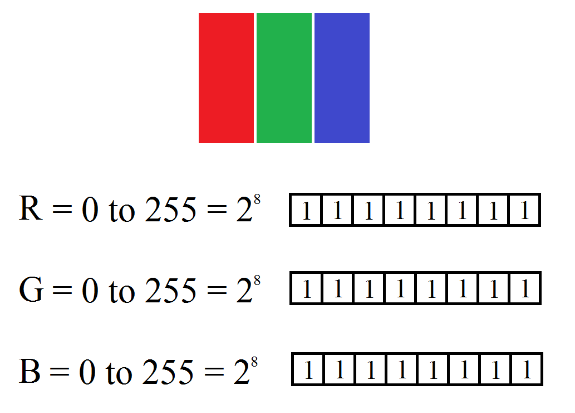
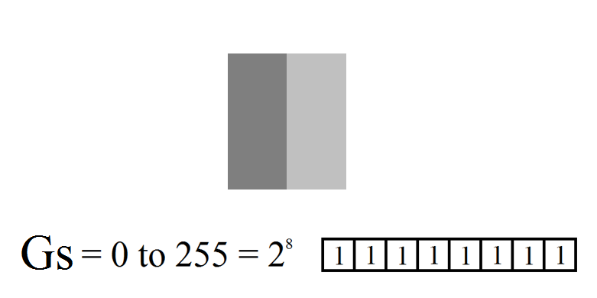
1. **PROBLEM STATEMENT**

Face recognition brings in several problems completely unique to this domain and which make it one of the most challenging in the group of machine learning problems.

Illumination problem – due to the reflexivity of human skin, even a slight change in the illumination of the image can widely affect the results. The image captured needed to be of a face that was looking almost directly at the camera, with little variance of light or facial expression from the image in the database. In most instances, if the images were not taken in a controlled environment, even the smallest changes in light or orientation could reduce effectiveness of the system, so they couldn't be matched to any face in the database, leading to a high rate of failure.

Pose changes – any rotation of the head of a person will affect the performance.

Time delay – of course that due to the aging of the human individuals, the database has to be regularly updated. Hence large samples of facial image (or train set) have to be fed to the system to make a relative comparison with the source more accurate. Also underlying with the concept of individual comparison with each train set for a single identity when consisting of a large sample size would be redundant and consume more time and computational memory as well.

Color Intensities – Also working with only image intensities, meaning the RGB pixel values in every single pixel in the image, made feature calculation rather computationally expensive and therefore slow on most platforms. Hence the average of RGB intensities must be taken to convert to greyscale for comparisons.  

~ 16.77 million colors 256 colors

Fig: A single 8-bit Color and Greyscale pixel

1. **SCOPE**

The dawn of digital revolution has generated a multitude of interrelated fields. So on, the application sector of Face Recognition covers a great deal of area alongside:

* **Digital Signature:**

Face Recognition is used in system that encrypt data and can be compared to other biometrics such as fingerprint or eye iris recognition systems that provide access. This can help to keep the confidential documents secured and prevent unauthorized access.

* **Forensic Analysis:**

The description of suspect or victim can be fed into system to find the closest match to gain further data about the subject. This is also a widely used application to crack down criminal cases effectively and efficiently.

* **Security Surveillance:**

Surveillance (or CCTV) cameras are often used in many different places, and in case of any incident, the footage is reviewed. The real time face recognition system can be implemented to track the individual face and give description about their identity.

* **Artificial Intelligence:**

AI or Artificial Intelligence has been marked as a rapid growing field inside computer science with implementation of DNN. Among the categories of AI requisites lies Computer Vision, which describes the ability for computer to see and distinguish objects and persons alike humans.

* **Autonomous Systems:**

This field includes self-driving cars and self-operating robots. The autonomous system can generally use Face Recognition to gain identity of the subject and use it for security purpose as well.

1. **LITERATURE REVIEW**

In 1960s, when Woody Bledsoe, Helen Chan Wolf, and Charles Bison developed face recognition using computer. In 1996 A.D., Bledsoe 1968a was tested at Stanford Research Institute by Peter Hart in 1996 AD with 2000 photographs. Moreover, the result was majorly acceptable. In 1997, the system developed by Christoph von der Malsburg and graduate students of the University of Bochum in Germany, by United States Army Research Laboratory.

In 2001, Paul Viola and Michael Jones developed an efficient algorithm for Face Detection. Later, Object recognition is a process for identifying a specific object in a digital image or video. Object recognition algorithms rely on matching, learning, or pattern recognition algorithms using appearance-based or feature-based techniques. Common techniques include edges, gradients, Histogram of Oriented Gradients (HOG), Haar wavelets, and linear binary patterns. Object recognition is useful in applications such as video stabilization, automated vehicle parking systems, and cell counting in bioimaging.

Basically, the process of recognizing a face in an image has two phases:

* Face detection – detecting the pixels in the image which represent the face. There are several algorithms for performing this task, one of these “Haar Cascade face detection” will be used later in the example, however not explained in the article.
* Face recognition – the actual task of recognizing the face by analyzing the part of the imaged identified during the face detection phase.

Appearance based statistical methods are methods which use statistics do define different ways how to measure the distance between two images. In other words, they try to find a way to say how similar two faces are to each other. There are several methods which fall into this group. The most significant are:

* Principal Component Analysis (PCA)
* Linear Discriminant Analysis
* Independent Component Analysis
* Gabor Filters – filters commonly used in image processing, that have a capability to capture important visual features. These filters are able to locate the important features in the image such eyes, nose or mouth. This method can be combined with the previously mentioned analytical methods to obtain better results.
* Neural Networks are simulating the behavior of human brain to perform machine learning tasks such as classification or prediction. In our case we need the classification of an image. The explication of Neural Networks would take at least one entire article (if not more). Basically, Neural Network is a set of interconnected nodes. The edges which are between the nodes are weighted so the information which travels between two nodes is amplified. The information travels from set of input nodes, across a set of hidden nodes to a set of output nodes. The developer has to invent a way to encode the input (in this case an image) to a set of input nodes and decode the output (in this case a label identifying the person) from the set of output points.

Commonly used method is to take one node for each pixel in the image on the input side of the network and one node for each person in the database on the output side as illustrated on the following image:

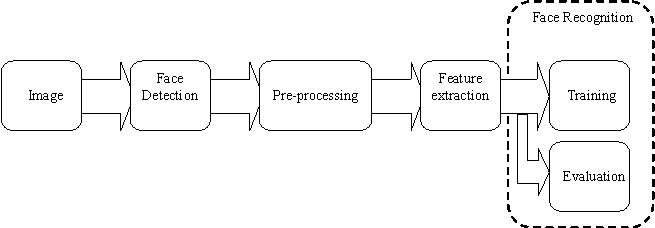


Fig: Face Recognition Process Steps

* 1. **Computer Vision**

As a scientific discipline, computer vision is concerned with the theory and technology for building artificial systems that obtain information from images or multi-dimensional data. A significant part of artificial intelligence deals with planning or deliberation for system which can perform mechanical actions such as moving a robot through some environment. This type of processing typically needs input data provided by a computer vision system, acting as a vision sensor and providing high-level information about the environment and the robot. Other parts which sometimes are described as belonging to artificial intelligence and which are used in relation to computer vision is pattern recognition and learning techniques.

Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images, and extraction of high-dimensional data from the real world in order to produce numerical or symbolic information, e.g., in the forms of decisions. Understanding in this context means the transformation of visual images (the input of the retina) into descriptions of the world that can interface with other thought processes and elicit appropriate action. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory.

As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. The image data can take many forms, such as video sequences, views from multiple cameras, or multi-dimensional data from a medical scanner. As a technological discipline, computer vision seeks to apply its theories and models for the construction of computer vision systems. Sub-domains of computer vision include: Scene reconstruction, Event detection, Video tracking, Face and Object Recognition, 3D pose estimation, Learning, Indexing, Motion estimation, and Image restoration.

* 1. **Open CV Overview**

OpenCV is an open source computer vision and machine learning software library built by Intel Corporation, Willow Garage, Itseez. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code. The library is used extensively in companies, research groups and by governmental bodies as well.

OpenCV also supports the deep learning frameworks TensorFlow, Torch/PyTorch and Caffe. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms.

* 1. **Haar Like Features**

Haar-like features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets, which is a sequence of rescaled "square-shaped" functions which together form a wavelet that allows a target function over an interval to be represented in terms of an orthonormal basis, and which were used in the first real-time face detector.

The problem with varying illumination of face on screen was addressed by Haar-like features, developed by Viola and Jones on the basis of the proposal by Papageorgiou et. al in 1998. A Haar-like feature considers neighboring rectangular regions at a specific location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. This difference is then used to categorize subsections of an image.

An example of this would be the detection of human faces. Commonly, the areas around the eyes are darker than the areas on the cheeks. One example of a Haar-like feature for face detection is therefore a set of two neighboring rectangular areas above the eye and cheek regions.

* 1. **Cascade Classifier**

Cascading is a particular case of ensemble learning based on the concatenation of several Classifiers, using all information collected from the output from a given classifier as additional information for the next classifier in the cascade. Unlike voting or stacking ensembles, which are multi-expert systems, cascading is a multistage one.

Cascading Classifiers are trained with several hundred "positive" sample views of a particular object and arbitrary "negative" images of the same size. After the classifier is trained it can be applied to a region of an image and detect the object in question. To search for the object in the entire frame, the search window can be moved across the image and check every location for the classifier. This process is most commonly used in image processing for object detection and tracking, primarily facial detection and recognition.

The cascade classifier consists of a list of stages, where each stage consists of a list of weak learners. The system detects objects in question by moving a window over the image. Each stage of the classifier labels the specific region defined by the current location of the window as either positive or negative – positive meaning that an object was found or negative means that the specified object was not found in the image.

If the labelling yields a negative result, then the classification of this specific region is hereby complete and the location of the window is moved to the next location.   
If the labelling gives a positive result, then the region moves of to the next stage of classification. The classifier yields a final verdict of positive, when all the stages, including the last one, yield a result, saying that the object is found in the image.

A true positive means that the object in question is indeed in the image and the classifier labels it as such – a positive result. A false positive means that the labelling process falsely determines, that the object is located in the image, although it is not. A false negative occurs when the classifier is unable to detect the actual object from the image and a true negative means that a non-object was correctly classifier as not being the object in question.

In order to work well, each stage of the cascade must have a low false negative rate, because if the actual object is classified as a non-object, then the classification of that branch stops, with no way to correct the mistake made. However, each stage can have a relatively high false positive rate, because even if the n-th stage classifies the non-object as actually being the object, then this mistake can be fixed in n+1-th and subsequent stages of the classifier.

* 1. **Haar Cascades**

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier. Then we need to extract features from it. For this, haar features shown in below image are used. They are alike convolutional kernel. Each feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle.



Fig: Haar Feature

Now all possible sizes and locations of each kernel is used to calculate plenty of features. A 24x24 window results over 160000 features. For each feature calculation, we need to find sum of pixels under white and black rectangles. To solve this, the integral images are introduced. It simplifies calculation of sum of pixels, how large may be the number of pixels, to an operation involving just four pixels.   
It although is irrelevant for practical use, we hence select the best features out of 160000+ features by using Adaboost.

We apply each and every feature on all the training images. For each feature, it finds the best threshold which will classify the faces to positive and negative. Although, there will be errors or misclassifications, we select the features with minimum error rate, which means they are the features that best classifies the face and non-face images. Then again same process is done to calculate new error rates and new weights. The process is continued until required accuracy or low error rate is achieved or required number of features are found.

Final classifier is a weighted sum of these weak classifiers. It is called weak because it alone can't classify the image, but together with others forms a strong classifier. The paper claims that even 200 features provide detection with 95% accuracy. Their final setup had around 6000 features.

It is a little inefficient and time consuming but authors solve this by checking if a window is a face region on not. If not, it will discard and won’t process it again. In an image, most of the image region is non-face region. So it is a better idea to have a simple method to check it. It instead focuses on region where there can be a face. This way, we can find more time to check a possible face region.

For this they introduced the concept of Cascade of Classifiers. Instead of applying all the 6000 features on a window, group the features into different stages of classifiers and apply one-by-one. Normally first few stages will contain very less number of features, if a window fails the first stage, it won’t consider remaining features on it. If it passes, apply the second stage of features and continue the process.   
On an average, 10 features out of 6000+ are evaluated per sub-window. This is the process how Viola-Jones face detection works.

* 1. **DIAGRAM**

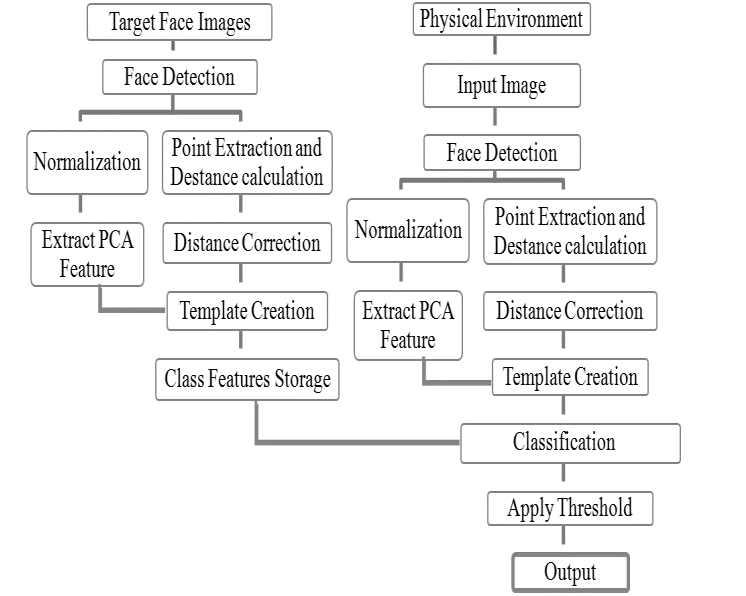
****

Fig: Block Diagram for Face Recognition

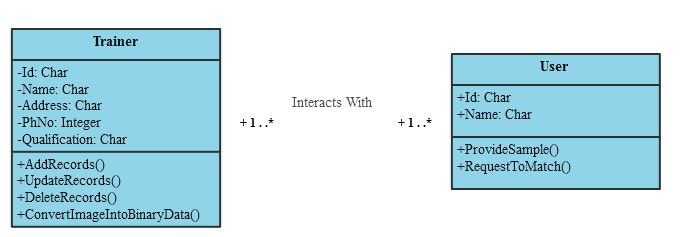
****

Fig: Class Diagram of Face Recognition

1. **METHODOLOGY**

In the beginning of the 1970's, face recognition was treated as a 2D pattern recognition problem. The distances between important points where used to recognize known faces, e.g. measuring the distance between the eyes or other important points or measuring different angles of facial components. But it is necessary that the face recognition systems to be fully automatic. Face recognition is such a challenging yet interesting problem that it has attracted researchers who have different backgrounds: psychology, pattern recognition, neural networks, computer vision, and computer graphics. The following methods are used to face recognition.

Feature-based (structural) Methods: In this method local features such as eyes, nose and mouth are first of all extracted and their locations and local statistics (geometric and/or appearance) are fed into a structural classifier. A big challenge for feature extraction methods is feature "restoration", this is when the system tries to retrieve features that are invisible due to large variations, e.g. head Pose when we are matching' a frontal image with a profile image. Distinguishes between three different extraction methods:

I. Generic methods based on edges, lines, and curves

II. Feature-template-based methods

III. Structural matching methods

1. **OUTPUT**

The screenshot of the partial output from the currently marked development until Face Detection phase of the program is as given below:

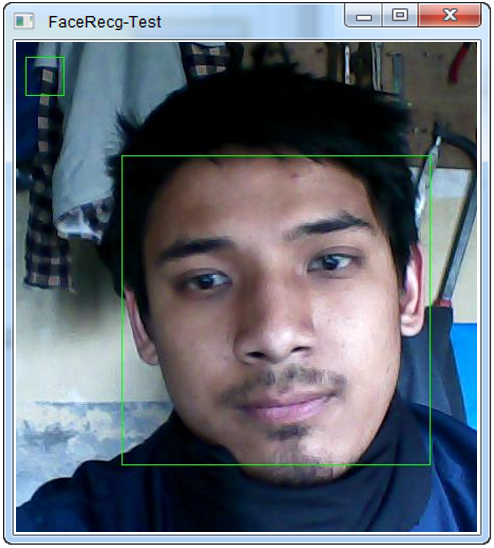


Fig: Output of Face Recognition (Face Detection)

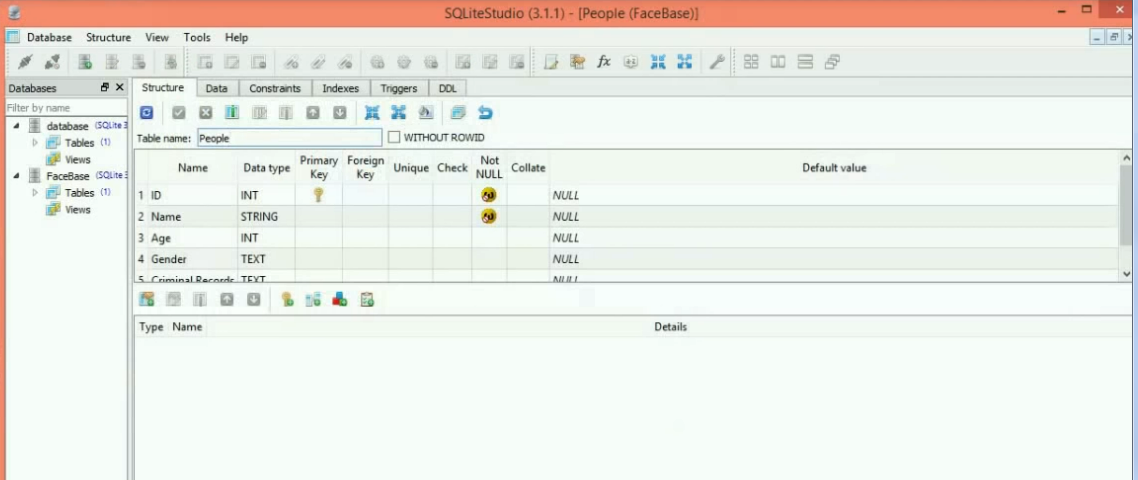


Fig: Face Database of SQLiteStudio under construction

1. **SCHEDULE**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Month**  **Work** | **August** | | | | **September** | | | | **October** | | | | **November** | | | | **December** | | | | **January** | | | |
| **Design Specification** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Mathematical Analysis** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Coding Implementation** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Testing and Debugging** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

Gantt Chart of Work Routine

1. **CONCLUSION**

Hence, for the current update of the program design, the Face Detection part was completed with additional database for Face Recognition remaining under construction.

1. **REFERENCE**

[*https://blog.octo.com/en/basics-face-recognition/*](https://blog.octo.com/en/basics-face-recognition/)[Site Visited on: 14th January, 2018]

Reinhard Klette (2014), *Concise Computer Vision*, Springer,   
ISBN 978-1-4471-6320-6

Linda G. Shapiro; George C. Stockman (2001), *Computer Vision*, Prentice Hall*,*   
ISBN 0-13-030796-3

Tim Morris (2004), *Computer Vision and Image Processing,* Palgrave Macmillan, ISBN 0-333-99451-5

Bernd Jähne; Horst Haußecker (2000), *Computer Vision and Applications*, *A Guide for Students and Practitioners,* Academic Press. ISBN 0-13-085198-1

David A. Forsyth; Jean Ponce (2003), *Computer Vision*, *A Modern Approach*,   
Prentice Hall, ISBN 0-13-085198-1

Viola and Jones, "*Rapid object detection using a boosted cascade of simple features*", Computer Vision and Pattern Recognition, 2001

Papageorgiou, Oren and Poggio, "*A general framework for object detection*", International Conference on Computer Vision, 1998

[*https://docs.opencv.org/3.3.0/d7/d8b/tutorial\_py\_face\_detection.html*](https://docs.opencv.org/3.3.0/d7/d8b/tutorial_py_face_detection.html)[Site Visited on: 14th January, 2018]