# A - 02 Major Project Documentation

*by* J SATWIK 221710301025

**Submission date:** 07-May-2021 01:06PM (UTC+0530)

**Submission ID:** 1580364467

**File name:** 5119_J_SATWIK_221710301025_A_-_02_Major_Project_Documentation_94915_1116423961.pdf (4.95M)

**Word count:** 3051

**Character count:** 15119

# Multi-Face Detection

**Major Project submitted in partial fulfillment of the requirements for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by**

**J. Satwik (221710301025)**

**P. Shyam (221710301041)**

**Pawan Kalyan (221710301010)**

**S. Avinash (221710301052)**

**Under the esteemed guidance of**

**Mrs. Chitta Hrudaya Neeharika**

**Assistant Professor**


**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GITAM**

**(Deemed to be University)**

**HYDERABAD**

**MAY 2021**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# GITAM

**(Deemed to be University)**

## DECLARATION

We hereby declare that the major project entitled "**Multi-Face Detection**" is an original work done in the Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University) submitted in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering. The work has not been submitted to any other college or university for the award of any degree or diploma.

Date:

**Name**

**J. Satwik (221710301025)**

**P. Shyam (221710301041)**

**Pawan Kalyan (221710301010)**

**S. Avinash (221710301052)**

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

## GITAM

**(Deemed to be University)**


# CERTIFICATE

This is to certify that the Major Project entitled "**Multi-Face Detection**" is submitted by **J. Satwik** (221710301025), **P. Shyam** (221710301041), **Pawan Kalyan** (221710301010), and **S. Avinash** (221710301052) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering. The Major Project has been approved as it satisfies the academic requirements.

**Guide**

**Mrs. Chitta Hrudaya Neeharika**

**Assistant Professor**

**Dept. of Computer Science and Engineering**

# ACKNOWLEDGMENT

Our Major Project would not have been successful without the help of several people. We would like to thank the personalities who were part of our project in numerous ways, those who gave us outstanding support from the birth of the project.

We are extremely thankful to our honorable Pro-Vice-Chancellor, **Prof. N. Siva Prasad,** for providing the necessary infrastructure and resources for the accomplishment of our project.

We are highly indebted to **Prof. N. Seetharamaiah**, Principal, School of Technology, for his support during the tenure of the project.

We are very much obliged to our beloved **Prof. S. Phani Kumar**, Head of the Department of Computer Science & Engineering, for providing the opportunity to undertake this project and encouragement in the completion of this project.

We hereby wish to express our deep sense of gratitude to **Dr. S Aparna**, Assistant Professor, Department of Computer Science and Engineering, School of Technology, and to **Mrs. Chitta Hrudaya Neeharika**, Assistant Professor, Department of Computer Science and Engineering, School of Technology, for the esteemed guidance, moral support and invaluable advice provided by them for the success of the project.

Sincerely,

J. Satwik (221710301025)

P. Shyam (221710301040)

Pawan Kalyan (221710301010)

S. Avinash (221710301052)

# TABLE OF CONTENTS

# ABSTRACT

Face Detection is a method used to detect faces using images or video surveillance cameras. It can be used to identify criminals equipped with city monitoring. This can also be used in crowded areas in order to detect multiple faces.

We can perform this task by using various methods such as convolution methods, data augmentation. By replacing the existing algorithm, we can use the MTCNN algorithm, which is very effective in detecting faces. An accuracy of about 92% can be achieved using the MTCNN algorithm. The face quality is increased by using this algorithm. It can also be used to increase the robustness of face detection and improve accuracy.

# LIST OF FIGURES

# CHAPTER 1

# INTRODUCTION

## 1.1. DIGITAL IMAGE PROCESSING

It is used for detecting objects in an image or performing processes such as noise removal, features extraction, etc.

We also need to differentiate between which object is the background and which object is acting as a shadow and find the required object. Generally, humans can easily detect objects using their brain, but for a computer or a machine to perform such an operation, it requires programming and lots of processing in order to detect an object.

An image is generally represented in two-dimensional arrays if it is a black & white image, but if it is a colored image, then it is represented as three-dimensional arrays.

## 1.2. BASICS OF IMAGE PROCESSING

## FUNDAMENTALS OF DIGITAL IMAGE

### 1.2.1. IMAGE

An image is a two – dimensional array containing the pixel values of the picture. An image can be a photograph, screen, or a statue. Images can be captured using cameras, lenses, telescopes, microscopes, etc. An image can be a map, graph, a pie chart, or a painting.

**Fig 1.1: Color Image & Gray Scale Image**

An image consists of pixels in a rectangular grid called a matrix. An image consists of height and width, which are defined in pixels. Each pixel have a size fixed. Each pixel represents color and brightness.

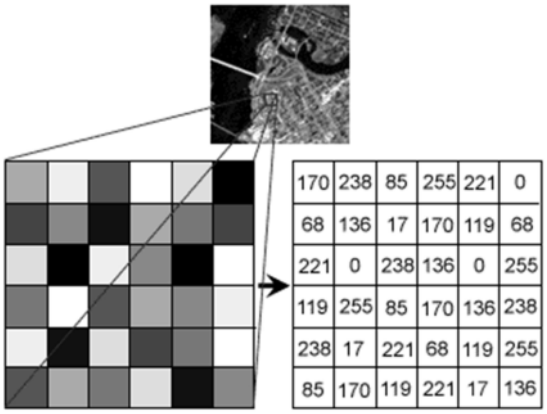


**Fig 1.2: Gray Scale Image Pixel Values**

Each pixel consists of color, and is a 32-bit integer. Every 8 bits represents colors intensity level of red, green, blue, and transparency.


**Fig 1.3: Bit Representation of an Image**

### 1.2.2. IMAGE FILE SIZE

Size of a file can be expressed in terms of bytes which depend on the depth of pixels, the number of pixels of the image and color. If we have more number of rows and columns, then that increases the image resolution and also the file size. 256 colors are stored in a 8-bit pixel, and 16 million colors in a 24-bit pixel. In order to decrease the file size, we can use image compression. Large image files are produced through high-resolution cameras. Different image file formats have been developed in order to store images of different pixels and different sizes and resolutions.

### 1.2.3. IMAGE FILE FORMATS

Different file formats are used to store images. There are different types of image file types; some of them include JPEG, PNG and GIF formats.

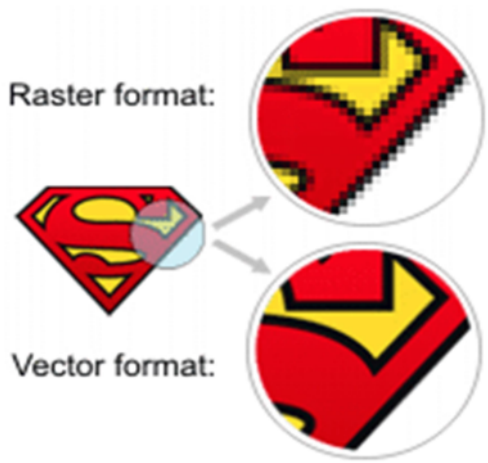


**Fig 1.4: Vector & Raster Format representation**

Metafile is another file format which includes both raster and vector information. Most of the Windows applications use this type of file format which then convert them into their native file formats.

### 1.2.4. FUNDAMENTAL STEPS IN IMAGE PROCESSING

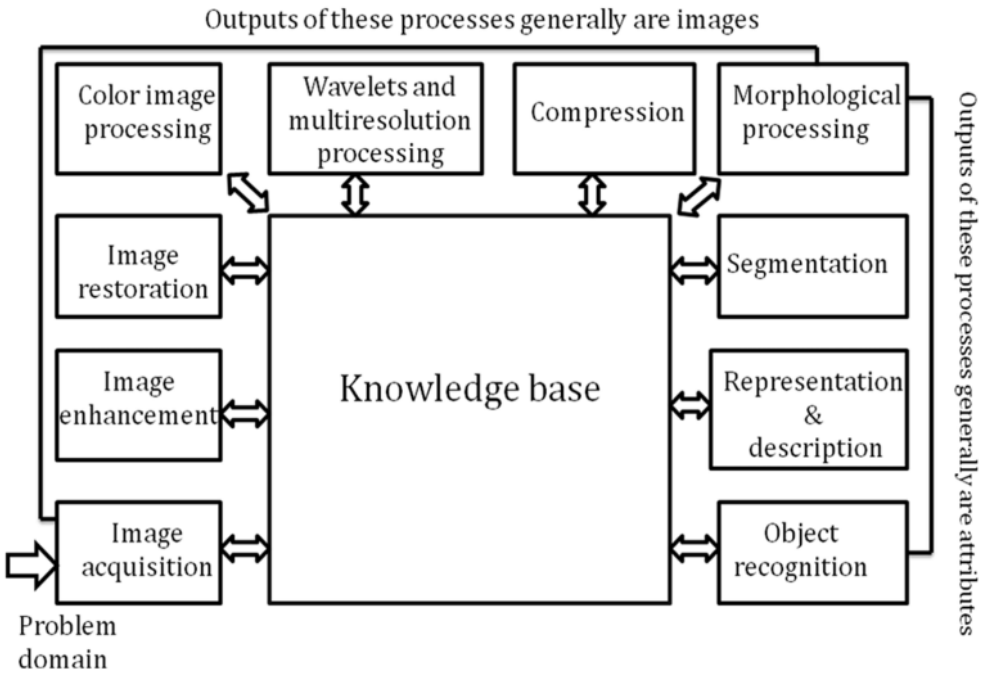


**Fig 1.5: Basic Steps of Image Processing**

- **IMAGE ACQUISITION**

It is the process of getting an image from a source for processing. It is the first step in the image processing workflow. The image that is acquired is unprocessed.

- **IMAGE ENHANCEMENT**

It helps in improving the quality and content of the image before it is processed from the original data.

**Fig 1.6: Image Enhancement process from Gray Scale Image to Color Image**

- **IMAGE RESTORATION**

Image Restoration is used to convert a noisy image into a clear and clean image. The noisiness in the image can be due to blurriness, mis-focus etc.


**Fig 1.7: Noise Image → Image Enhancement**

- **COMPRESSION**

Compression is used to convert the image or file into a small file which occupies less space than the original file or image.

- **MORPHOLOGICAL PROCESSING**

Morphological processing is a set of operations done on an image based on its shapes.

- **SEGMENTATION**

Segmentation is used for partitioning or dividing an image into multiple segments; it is generally used to find boundaries and objects in an image.

- **OBJECT RECOGNITION**

To detect an object in the real world from an image or a video, we use Object recognition techniques.

## 1.3. CLASSIFICATION OF IMAGES

### 1.3.1. GRAYSCALE IMAGE

Gray scale images are composed of shades of black and white. They are also called as monochromatic images.

### 1.3.2. BINARY IMAGE

A binary image is said to have only two values for each pixel those are White and Black. Each pixel is a single bit either as 1's or 0's. Binary images are used in various applications such as laser printers, fax machines, and some input and output devices.

### 1.3.3. COLOR IMAGE

A color image has different pixel values. Each pixel value determines different color intensities. The value of each pixel is defined by primarily three colors those are Red, Green, and Blue. The colors can be represented using their RGB values; for example, black can be represented as (0, 0, 0), whereas pink can be represented as (255, 0, 255).

A colored image contains 256 * 256 * 256 = 16.8 million colors. Colored image is a three-dimensional array of color values and pixels.

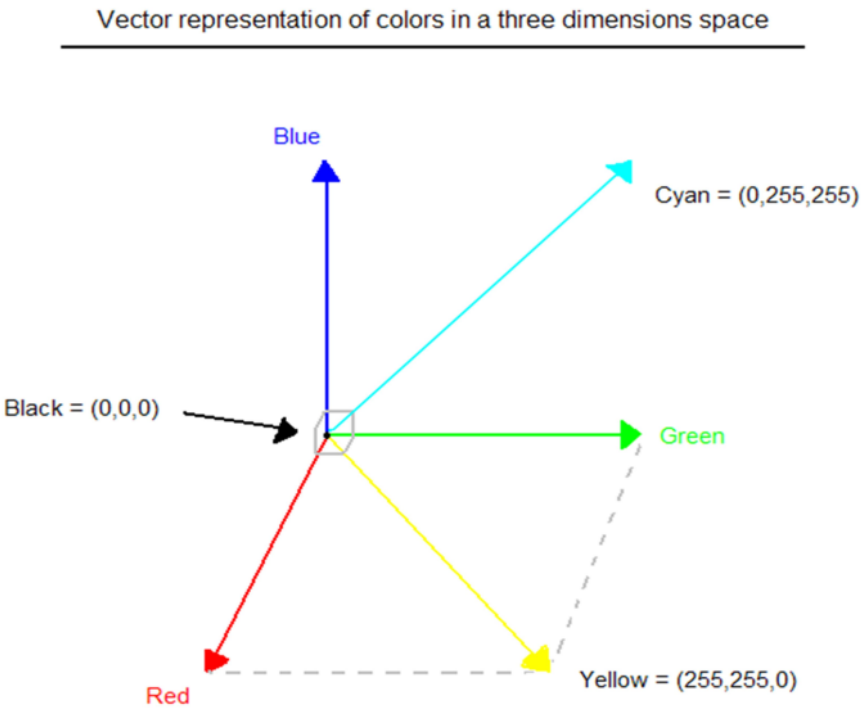


**Fig 1.8: Hue Saturation Process of RCB Scale Image**

# CHAPTER 2

# LITERATURE SURVEY

**Michael J Jones and Paul Viola. 2004. "Robust real-time face detection". IJCV, pp. 137-154, vol. 23, no. 2.**

This paper describes how we can detect faces and provides a framework that rapidly processes the images and gives high rate of detection. They described three methods to detect the faces rapidly.

The first is the "**Integral Image**", which helps to quickly compute the features used by the detector or program.

The second is a simple and efficient classifier using AdaBoost algorithm, which uses large set of features to select a small number of features.

The third one is combining all classifiers in a cascade which will quickly discard background images and just focusing on regions such as face.

This paper has proved better than the previous versions or systems and has given better results and accuracy. It was implemented on normal desktop which has the face detection capacity of 15 frames per second.

**Shen, H, Li, Lin, X., Z. L., "A convolutional neural network for face detection." *computer vision and pattern recognition*. et al., 2015, 5325-5334.**

It is difficult to detect a face in unconstrained conditions, which may include variations in pose, illumination, scale, expressions, and makeup, etc.

By using Convolution Neural Network (CNN), there has been improvement in the accuracy and performance of the face detector.

This paper mainly focuses on detecting faces using the latest Convolution Neural Networks (CNN). The paper also provides future guidelines in this field.

# CHAPTER 3

# DESIGN

## 3.1. SOFTWARE REQUIREMENTS

Software requirements include operating systems, design constraints, graphics requirements, and environment.

For this project the software requirements are

- Python IDLE
- Anaconda
- Numpy packages
- OpenCV packages
- Web Camera

## 3.2. HARDWARE REQUIREMENTS

- **Processor**: minimum Intel i3
- **Hard Disk**: minimum 250 GB
- **RAM**: minimum 4 GB
- **Operating System**: Windows, Linux

## 3.3. UML DIAGRAMS

### 3.3.1. SEQUENCE DIAGRAM

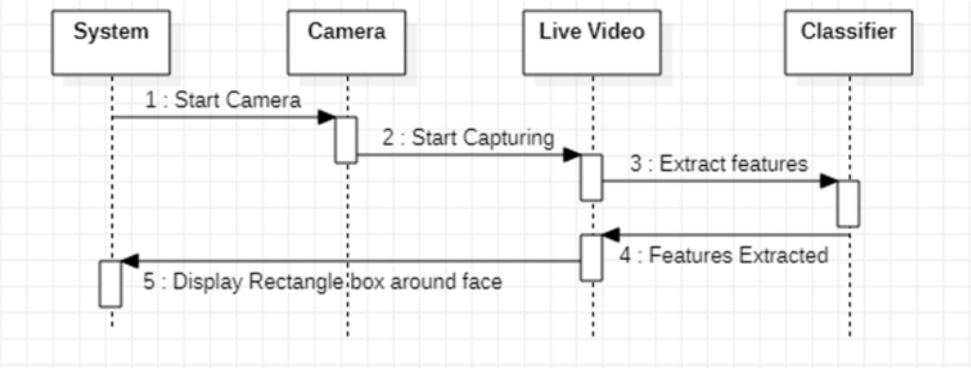


**Fig 3.1: Sequence Diagram**

A sequence diagram shows what happens first and what happens next in a time-based sequence. Sequence diagrams provide information to understand interfaces and class responsibilities.

### 3.3.2. USE CASE DIAGRAM

This diagram has actors representing the users and the steps that the actor has to follow to run or execute the program. It also represents how one function is dependent on the other.

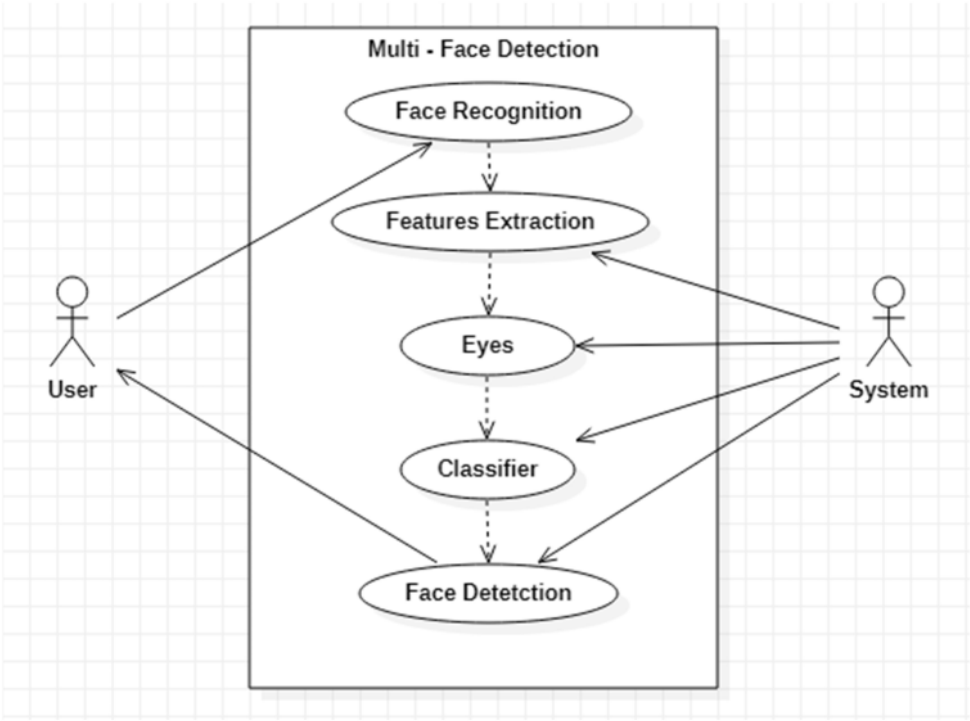


**Fig 3.2: Use Case Diagram**

# CHAPTER 4

# IMPLEMENTATION

## 4.1. FLOWCHART

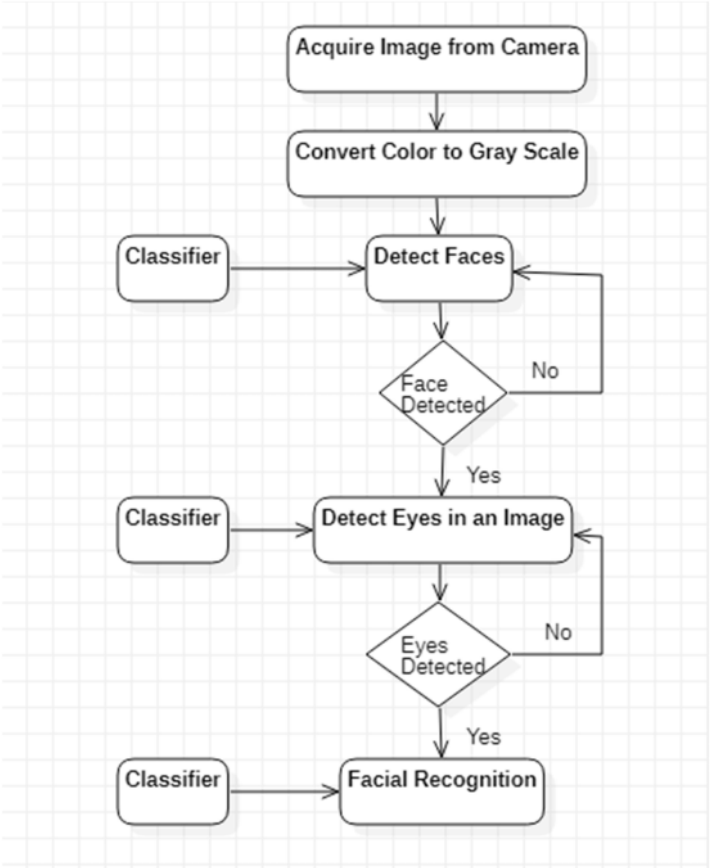


**Fig 4.1: Flow Chart**

## 4.2. CLASSIFIER

In this project, we have used the “Haar Cascade Classifier” to detect faces. This classifier is used to detect any object in real-time. It detects objects in an image and video.

This classifier uses positive images and negative images to detect objects in an image, where the positive image being the required object and negative images are other than the necessary object such as background, other objects, etc.

The classifier has four stages

### 4.2.1. CALCULATING HAAR FEATURES

Rectangular regions that are adjacent to each other are taken to perform the calculations. This calculation is done by adding the pixel intensities and then calculating the differences between the sums in each region.

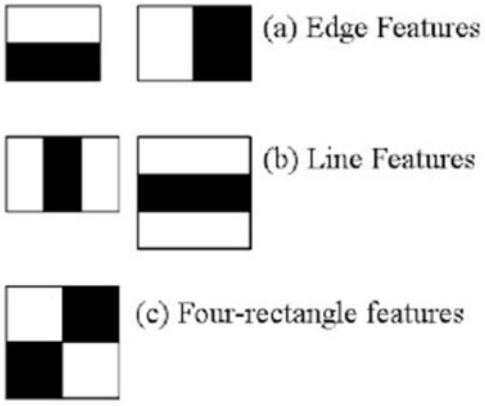


**Fig 4.2: Types of Haar Features**

### 4.2.2. CREATING INTEGRAL IMAGES

Calculating Haar features for a large image is difficult, so we use integral images. Integral images help in speeding up the process. Each pixel is not computed; instead, array references are made for each sub-rectangle.

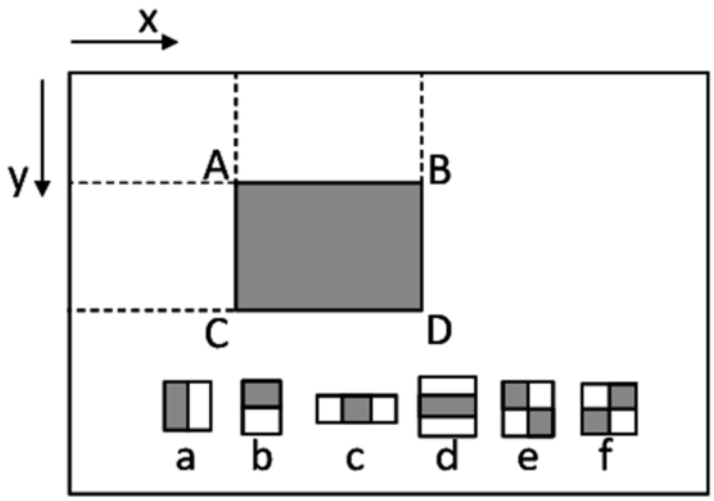


**Fig 4.3: Illustration of how an Integral image works**

### 4.2.3. USING ADABOOST

To detect features form an object we use AdaBoost. After selecting the best features, classifiers are trained to use them. All the weak classifiers are combined to create a strong classifier to detect objects.

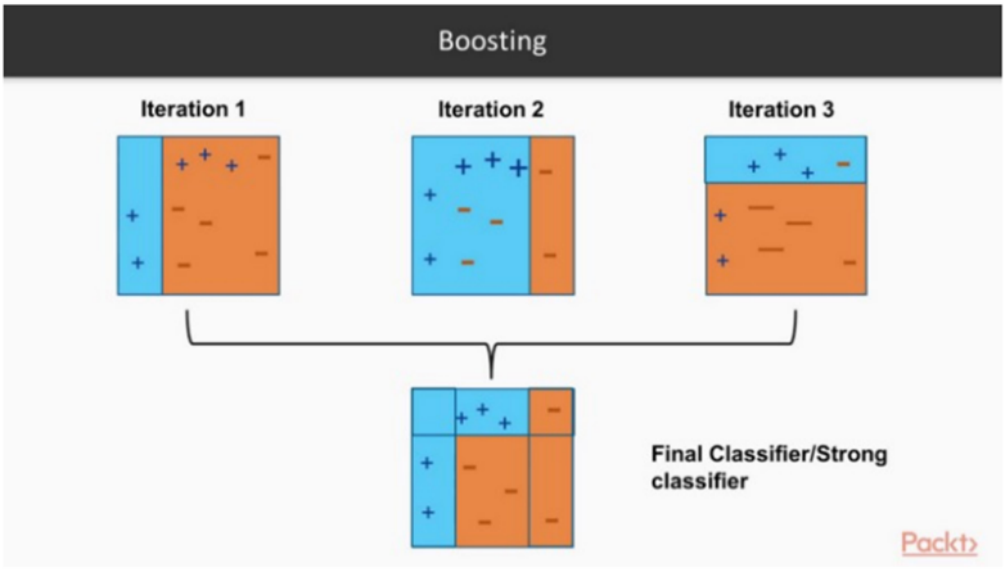


**Fig 4.4: Representation of Boosting Algorithm**

### 4.2.4. IMPLEMENTING CASCADING CLASSIFIERS

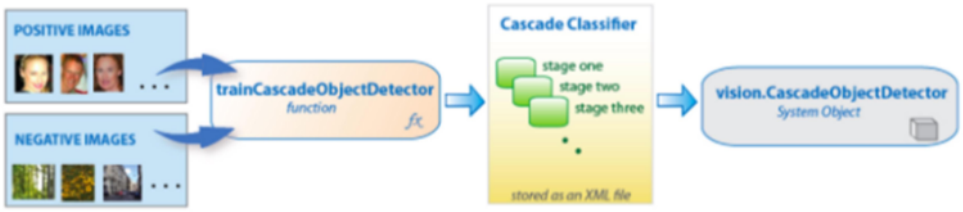


**Fig 4.5: Flowchart of Cascade Classifiers**

This is the last step where all the weak learners combine to a strong learner using cascading classifiers. If an object is found then it is labeled as positive and the remaining background images are labeled as negative.

Stages reject the negative areas and focus on detecting all the positive areas to find the object.

# CHAPTER 5

# TESTING

## 5.1. SOFTWARE TESTING

Testing can be said as the process of executing a program to find an error. The program should not contain any errors.

## 5.2. TYPES OF TESTING

- Unit Testing
- White Box Testing
- Beta Testing
- Black Box Testing etc.

### 5.2.1. UNIT TESTING

Unit testing is done on the smallest unit of program. Individual units are tested in this type of testing. Programmers perform such type of testing's.

### 5.2.2. WHITE BOX TESTING

How can we achieve the required output is the main focus in this type of testing. Generally used for verification.

### 5.2.3. BETA TESTIING

Beta testing is done by releasing the software to public but only to a limited number of people. This is done to check the real time problems that might arise during usage. Depending on the feedback given by the users the required changes are made to the software.

### 5.2.4. BLACK BOX TESTING

What would be the output is the main focus in this type of testing keeping aside the internal mechanism. Used for validation.

# CHAPTER 6

# RESULTS

To start the application, we need to open the Anaconda Prompt first.


**Fig 6.1: Anaconda Prompt**

Then in prompt, navigate to the folder containing the python file and execute the file using "**python file_name.py**".

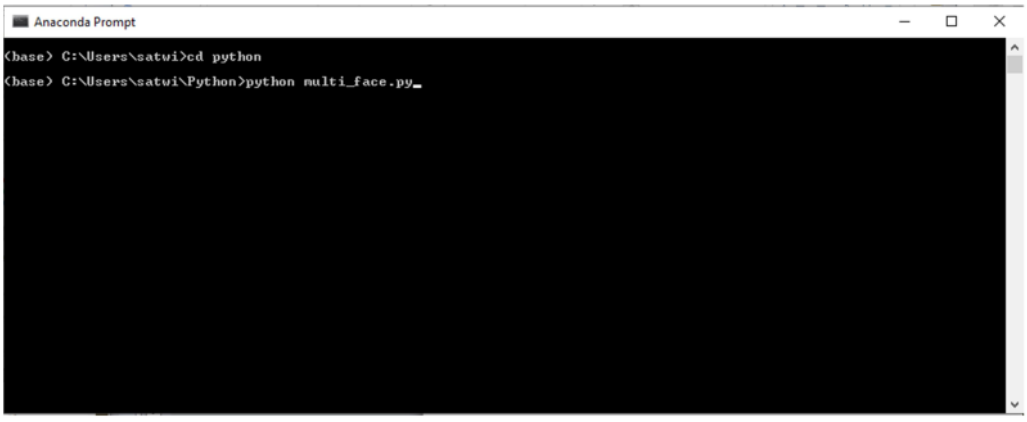


**Fig 6.2: File Navigation & Program Execution**

After successful execution of the code, a frame appears which captures images using the webcam.

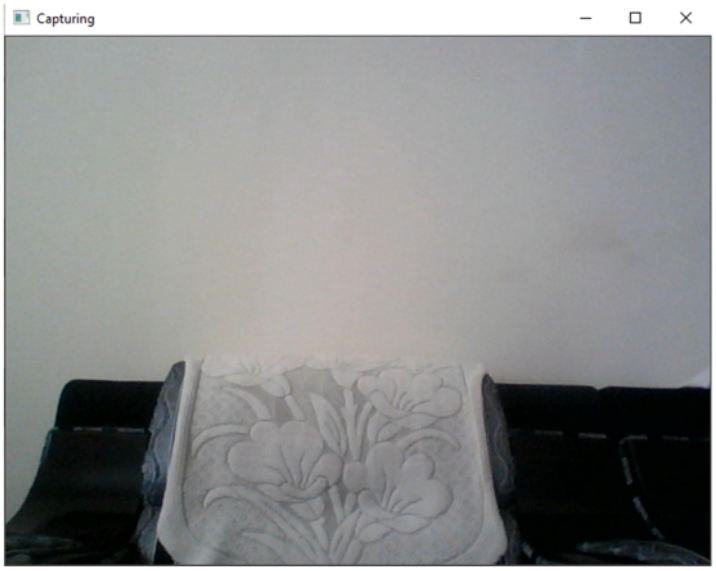


**Fig 6.3: Image Capturing**

When the window opens, it starts detecting faces in the frame. Once it successfully detects a face, a rectangular box is placed around the face.

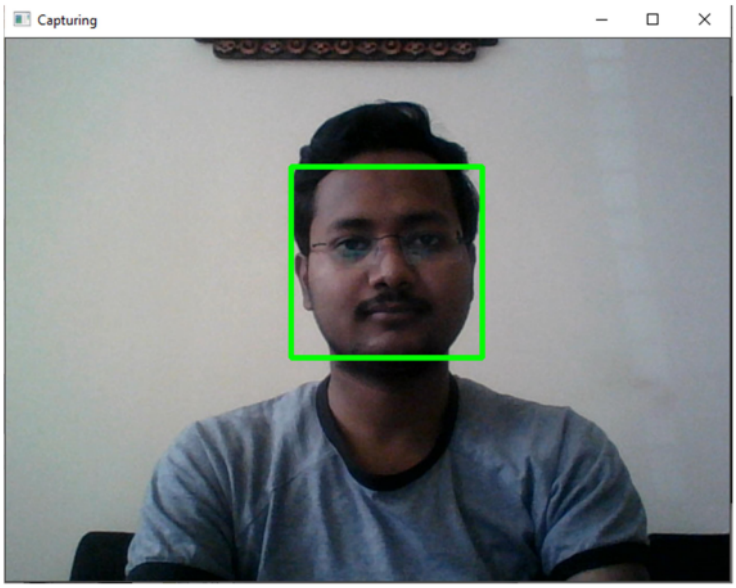


**Fig 6.4: Face Detection**

# CHAPTER 7

# CONCLUSION & FUTURE SCOPE

## 7.1. CONCLUSION

Face detection has been very popular for the past few years. Today, we can automatically verify identity for secure transactions, surveillance, and security tasks. High accuracy can be achieved using different algorithms, different methods, and systems.

Face detection has got various applications in several domains. Better accuracy and correct recognition of the face can be achieved using these algorithms and techniques.

This project helps in various applications that use video or images to monitor faces and meet security needs. We have discussed face detection, tracking, features extraction, etc.

## 7.2. FUTURE SCOPE

The future for face detection is very vast. In the future, we might see face detection being used for personal identification in place of passports, banks and each face may have a separate id to identify the person.

Face detection applications can be expanded and used in various fields:

- Such as tracking a person using his face, which can help find missing people or any criminal, etc.
- Using face detection, we can lip-read the face and convert them into text or audio format, which can help disabled people understand easily.
- Can be used in behavior monitoring in child care centers or old age homes or malls etc.
- Can be used in self-driving cars or any mode of transport to detect the driver's face and warn them if they are not alert and avoid accidents.

# REFERENCES


1. Michael J Jones, Paul Viola.**2004**. "*Robust real-time face detection*". **pp .137-154, vol. 23,** IJCV**, no. 2.**

2. Stan Z. Li, Shengcai Liao, and Anil K. Jain.. "*A fast and accurate unconstrained face detector*". TPAMI, **2016**.

3. X., Shen, H, Li, Z. L., Lin, et al. **2015**. "*A convolutional neural network for face detection. computer vision and pattern recognition*". **5325-5334**.

4. Jianxin Wu, S. Charles Brubaker, James M Rehg, Matthew D. Mullin and Jie Sun.. **2008**. "*On thedesign of cascades of boosted ensembles for face detection*". IJCV .

5. Xiaolin Hu, Xiu Li, Hongwei Qin and Junjie Yan. **2016**. "*Joint training of cascaded CNN for face detection. In CVPR.*"

# A - 02 Major Project Documentation

ORIGINALITY REPORT

| 17% | 15% | 2% | 15% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | Submitted to Gitam University<br>Student Paper | 10% |
| 2 | www.coursehero.com<br>Internet Source | 3% |
| 3 | Submitted to Nottingham Trent University<br>Student Paper | 1% |
| 4 | Submitted to University of Technology, Sydney<br>Student Paper | 1% |
| 5 | www.slideshare.net<br>Internet Source | 1% |
| 6 | Submitted to University of Bridgeport<br>Student Paper | <1% |
| 7 | ediss.uni-goettingen.de<br>Internet Source | <1% |
| 8 | Submitted to SASTRA University<br>Student Paper | <1% |
| 9 | baadalsg.inflibnet.ac.in<br>Internet Source | <1% |

| | | |
|---|---|---|
| 10 | hendisaleh.wordpress.com<br>Internet Source | <1% |
| 11 | tomfox.org<br>Internet Source | <1% |
| 12 | dspace.daffodilvarsity.edu.bd:8080<br>Internet Source | <1% |
| 13 | Lezi Wang, Xiang Yu, Dimitris N. Metaxas. "A Coupled Encoder-Decoder Network for Joint Face Detection and Landmark Localization", 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), 2017<br>Publication | <1% |
| 14 | hal.inria.fr<br>Internet Source | <1% |

| | | | |
|---|---|---|---|
| Exclude quotes | On | Exclude matches | Off |
| Exclude bibliography | On | | |