

**A**  
**Project Report**

**REAL TIME FACE RECOGNITION**

Submitted to  
**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES**  
**RK VALLEY**

*in partial fulfillment of the requirement for the award of the Degree of*

**BACHELOR OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE AND ENGINEERING**

Submitted by  
**BOYA NAGAMANI (R180373)**  
**CHINTHAPANDU VENKATA SUPRAJA (R180380)**

Under the Guidance of  
**Ms. S.RAJESWARI, Guest Faculty**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**  
**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES**

**(catering the Educational Needs of Gifted Rural Youth of AP)**

**R.K Valley, Vempalli(M), Kadapa(Dist) – 516330**

**2020 – 2024**

**RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOLOGIES**



**(A.P.Government Act 18 of 2008)**

**RGUKT-RKValley**

**Vempalli, Kadapa, Andhrapradesh - 516330.**

**CERTIFICATE OF PROJECT COMPLETION**

This is to certify that I have examined the thesis entitled as “**REAL TIME FACE RECOGNITION** ” submitted by **BOYA NAGAMANI (R180373), CHINTHAPANDU VENKATA SUPRAJA (R180380)** under our guidance and supervision for the partial fulfilment for the degree of Bachelor of Technology in computer Science and Engineering during the academic session February 2023 – July 2023 at RGUKT-RKVALLEY.

**Project Guide**

**Ms. S.Rajeswari ,**  
Guest Faculty in Dept of CSE,  
RGUKT-RK Valley.

**Head of the Department**

**Mr. N.Satyanandaram,**  
HOD in Dept of CSE,  
RGUKT-RK Valley.

**SIGNATURE OF GUIDE**

**SIGNATURE OF HOD**

# **RAJIV GANDHI UNIVERSITY OF KNOWLEDGE AND TECHNOLOGIES**

**(A.P.Government Act 18 of 2008) RGUKT-RK Valley**

**Vempalli, Kadapa, Andhrapradesh-516330.**

## **DECLARATION**

We, **BOYA NAGAMANI (R180373), CHINTHAPANDU VENKATA SUPRAJA (R180380)** hereby declare that the project report entitled “**REAL TIME FACE RECOGNITION** ” done under guidance of **Ms. S. Rajeswari** is submitted in partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering during the academic session February 2023 – July 2023 at RGUKT-RK Valley. I also declare that this project is a result of our own effort and has not been copied or imitated from any source. Citations from any websites are mentioned in the references. To the best of my knowledge, the results embodied in this dissertation work have not been submitted to any university or institute for the award of any degree or diploma.

Date :

Boya Nagamani-R180373

Chinthapandu Venkata Supraja-R180380

Place : RK Valley

## **ACKNOWLEDGEMENT**

I would like to express my deep sense of gratitude & respect to all those people behind the screen who guided, inspired and helped me crown all my efforts with success. I wish to express my gratitude to **S.Rajeswari** for her valuable guidance at all stages of study, advice, constructive suggestions, supportive attitude and continuous encouragement, without which it would not be possible to complete this project.

I would also like to extend our deepest gratitude & reverence to the Director of RGUKT, RK Valley **Dr. A.V.S.S Kumara Swami Gupta** and HOD of Computer Science and Engineering **Mr. N. Satyanandaram** for their constant support and encouragement.

**With Sincere Regards,**

**Boya Nagamani-R180373,**

**Chinthapandu Venkata Supraja-R180380,**

## **ABSTRACT**

The idea behind this technique involves using a cascade of classifiers to detect different features in an image. These classifiers are then combined into one strong classifier that can accurately distinguish between samples that contain a human face from those that don't.

The haar cascade classifier that is built into opencv has already been trained on a large dataset of human faces, so no further training is required. We just need to load the classifier from the library and use it to perform face detection on an input image.

This report contains how using python we have tried to implement one of the most important part of computer science, deep learning, which can be used for detecting or recognizing human faces. We have proposed a system that can help in recognizing a human face in real-time. This can be used for various purpose and various machine and smart devices

# Index

Page No

Certificate:.....	2
Declaration:.....	3
Acknowledgment:.....	4
Abstract:.....	5

Contents::

## Chapter 1:

Introduction.....	7-10
1.1 Motivation.....	
1.2 innovation of the project.....	
1.2 Objective of the Project.....	
1.3 Features.....	
1.3.1 applicatons.....	

## Chapter 2:

2 Technologies Used.....	11-13
2.1 python.....	
2.1.1 features of python.....	
2.2 Opencv.....	
2.2.1 features of opencv library.....	
2.3 face detection classifiers.....	
2.4 system architecture.....	

## Chapter 3:

Implementation:.....	14-20
3.1 problem statement.....	
3.2 Iplematation and Results.....	

## Chapter 4:

4.1	
Conclusion .....	21
4.2 Future Works.....	22

## Chapter 5:

References.....	23
-----------------	----

# **CHAPTER 1**

## **INTRODUCTION**

Face recognition is a technique that an electronic device like a computer can determine and recognize a human face by comparing the given sample of facial attributes of that person. This subject or topic, using unbiased facial data, computer-based face recognition as an analysis is a largely unexplained part of research study.

### **1.1 MOTIVATION:**

We developed this Face Detection Project for our Indian Army, Security CCTV Police and even Common Use of people to detect Face in single and group of peoples in Images.

Today In world Security is advancing to detect the terrorist and criminals easily in crowd. This Project will easily detect faces even old\blur Images with very less time of execution.

This Face Detection Code can be deployed and use in any application camera, security apps and website to found faces easily.

### **1.2 Innovation Idea of Project:**

1. It can Detect faces easily with less time of execution.
2. It can Detect Face in any quality of Images even blur.

3. It can Detect single, double or even multiple faces in any Image.
4. It can tell the number of Peoples/faces in Image.
5. It is fully automatic face recognition system
6. It is one of the best in today world Innovation as per less number of code use in the system.
7. It use Haar feature based Algorithm to detect which it makes very accurate and fast processing of face detection

### **1.3 Objective of Project :**

**The objectives of Face Recognition for Real-Time Applications are given below:**

- - to enhance the frame/sec for face recognition system, such that recognition is done in real time.
- Since human faces are so diverse, face detection models typically need to be trained on large amounts of input data for them to be accurate.
- thankfully, the opencv package comes with pre-trained models for face detection, which means that we don't have to train an algorithm from scratch. More specifically, the library employs a machine learning approach called haar cascade to identify objects in visual data.



## **1.4 Features:**

### **What is Computer Vision?**

Computer vision is a deep learning application that lies at the heart of this revolution. It allows computers to derive insight from visual input such as images and video files. Examples of computer vision include face detection, facial recognition, human pose estimation, and obstacle detection.

### **Applications of Computer Vision :**

#### **Surveillance**

Computer vision applications such as object recognition and pose estimation are often deployed in security devices to automate human surveillance.

#### **Retail**

Computer vision models can be deployed in retail outlets to track customers' eye position, body language, and movement around the store.

Retailers can then use these insights to improve the store's marketing strategy and customize product placement to drive sales.

#### **Autonomous Vehicles**

The field of autonomous driving has benefited tremendously from computer vision technologies.

Object detection models are deployed in vehicles to identify pedestrians, other vehicles, and animals on the road.

Computer vision applications can interpret stop signs and traffic lights, accurately estimate the distance between the vehicle and other objects, and avoid obstacles like potholes to ensure a safe driving experience.

## **1.5 Introduction to OpenCV:**

OpenCV is a computer vision library that supports programming languages like Python, C++, and Java.

OpenCV allows developers and non-mathematicians to build computer vision applications easily without having to code them from scratch. The library has over 2,500 algorithms that allow users to perform tasks like face recognition and object detection.

Developers and data practitioners at well-established organizations like Google, Microsoft, IBM, and Intel make extensive use of the OpenCV library, which is currently free for commercial use.

### **FEATURES:**

- Detect human faces in images with OpenCV in Python
- Perform real-time face detection in a live stream from a webcam
- Recognize and label celebrity faces in images

# **CHAPTER- 2**

## **REQUIREMENT ANALYSIS**

### **2.1 Python:**

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a generalpurpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems.

#### **2.1.1 features:**

Easy to learn and readable language. Python is extremely easy to learn

- Interpreted language.
- Dynamically typed language
- Open source and free
- Large standard library
- High-level language
- Object oriented programming language
- Large community support

### **2.2 opencv:**

Opencv is a great tool for image processing and performing computer vision tasks. It is an open-source library that can be used to perform tasks like face detection, objection tracking, landmark detection, and much more.

- Opencv is a huge open-source library for computer vision, machine learning, and image processing.

- Opencv supports a wide variety of programming languages like python, c++, java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human

### **2.2.1 features of opencv library**

Using opencv library, you can –

- Capture and save videos
- Process images (filter, transform)
- Perform feature detection
- Detect specific objects such as faces, eyes, cars, in the videos or images.
- Analyze the video, i.e., estimate the motion in it, subtract the background, and track objects in it.

### **2.3 Face detection classifiers:**

Face detection is performed by using classifiers. A classifier is essentially an algorithm that decides whether a given image is

positive(face) or negative(not a face). A classifier needs to be trained on thousands of images with and without faces.

Fortunately,

OpenCV already has two pre-trained face detection classifiers, which can readily be used in a program. The two classifiers are:

- Haar Classifier and
- Local Binary Pattern (LBP)

For this Project, Will only discuss the Haar Classifier for Face Detection as it best method for detection.

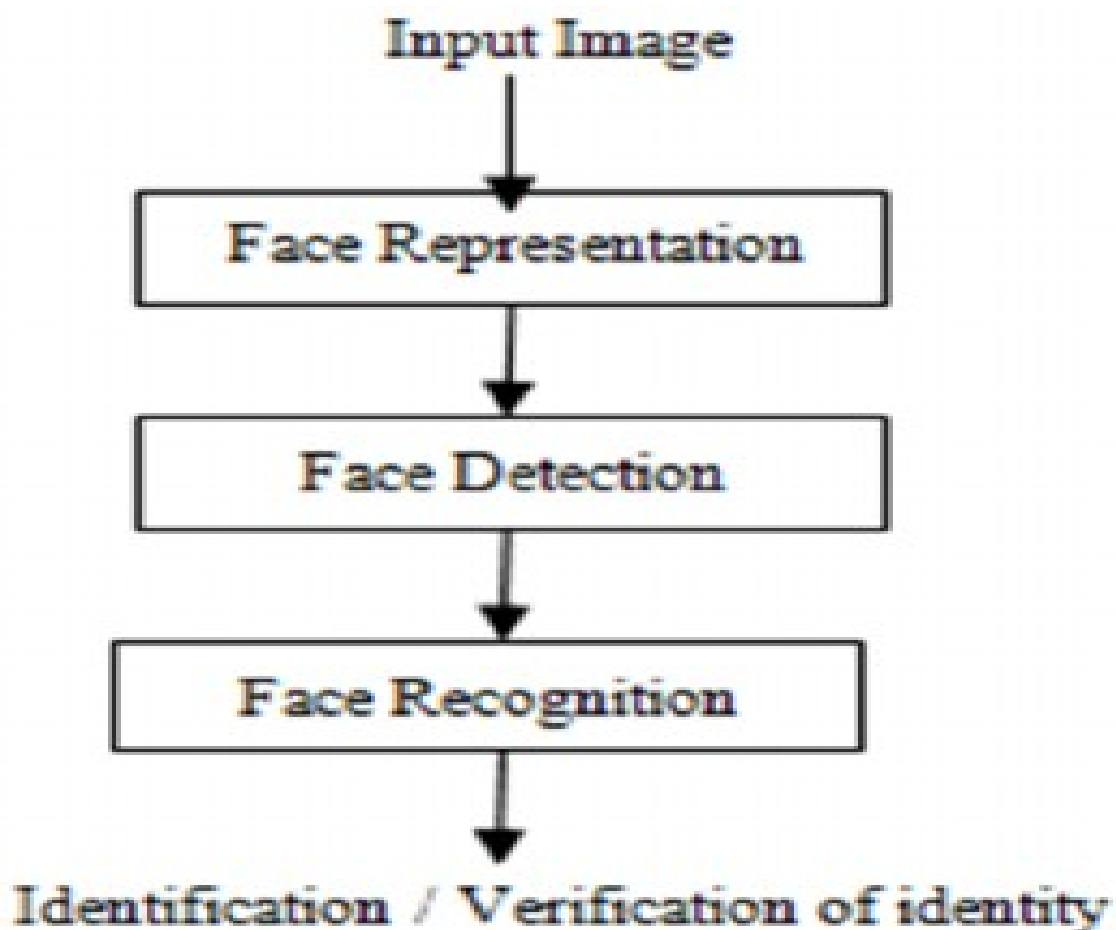
### **Haar feature-based cascade classifiers**

Haar-like features are digital image features used in object recognition.

This classifier is widely used for tasks like face detection in computer vision industry.

Haar feature-based cascade classifiers is an effectual machine learning based approach, in which a cascade function is trained using a sample that contains a lot of positive and negative images.

### **2.4 SYSTEM MODEL ARCHITECTURE:**



## CHAPTER -3

### IMPLEMENTATION

#### 3.1 Problem Statement:

The Problem statement of Face Recognition for Real-Time Applications are given below:

- To do face recognition in real time.
- Enhance the Speed i.e. frames/sec.
- Do recognition on high Camera resolution.

#### 3.2 Installation:

OpenCV-Python supports all the leading platforms like Mac OS, Linux, and Windows. It can be

installed in either of the following ways:

Packages for standard desktop environments (Windows, macOS, almost any GNU/Linux

distribution)

- **run pip install opencv-python** if you need only main modules
- **run pip install opencv-contrib-python** if you need both main and contrib modules

## **Step 1: Import the OpenCV Package**

Now, let's import OpenCV and enter the input image path with the following lines of code:

```
import cv2  
imagePath = 'input_image.jpg'
```

## **Step 2: Read the Image**

Then, we need to read the image with OpenCV's `imread()` function:

```
img = cv2.imread(imagePath)
```

This will load the image from the specified file path and return it in the form of a Numpy array.

## **Step 3: Convert the Image to Grayscale**

To improve computational efficiency, we first need to convert this image to grayscale before performing face detection on it:

```
gray_image = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

## **Step 4: Load the Classifier**

Let's load the pre-trained Haar Cascade classifier that is built into OpenCV:

```
face_classifier = cv2.CascadeClassifier(  
    cv2.data.harcascades + "haarcascade_frontalface_default.xml"  
)
```

## Step 5: Perform the Face Detection

We can now perform face detection on the grayscale image using the classifier we just loaded:

```
face = face_classifier.detectMultiScale(  
    gray_image, scaleFactor=1.1, minNeighbors=5,  
    minSize=(40, 40)  
)
```

### **1.minNeighbors:**

The cascade classifier applies a sliding window through the image to detect faces in it. You can think of these windows as rectangles.

### **2.minSize:**

Finally, the minSize parameter sets the minimum size of the object to be detected. The model will ignore faces that are smaller than the minimum size specified.

## **step 6: Drawing a Bounding Box**

Now that the model has detected the faces within the image, let's run the following lines of code to create a bounding box around these faces:

```
for (x, y, w, h) in face:  
    cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 4)
```

The face variable is an array with four values: the x and y axis in which the faces were detected, and their width and height. The above code iterates over the identified faces and creates a bounding box that spans across these measurements.



The parameter 0,255,0 represents the color of the bounding box, which is green, and 4 indicates its thickness.

### Step 7: Displaying the Image

To display the image with the detected faces, we first need to convert the image from the BGR format to RGB:

```
img_rgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
```

Now, let's use the Matplotlib library to display the image:

```
import matplotlib.pyplot as plt  
plt.figure(figsize=(20,10))  
plt.imshow(img_rgb)  
plt.axis('off')
```

## 3.3 RESULTS:

The above code should generate the following output:

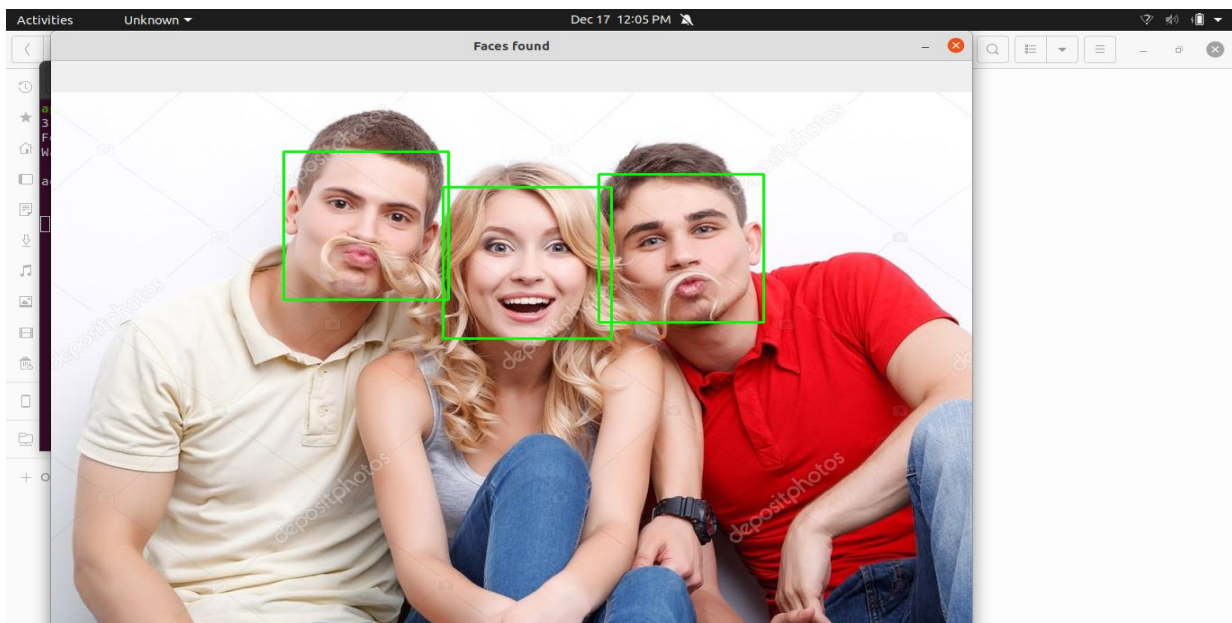


Fig.1

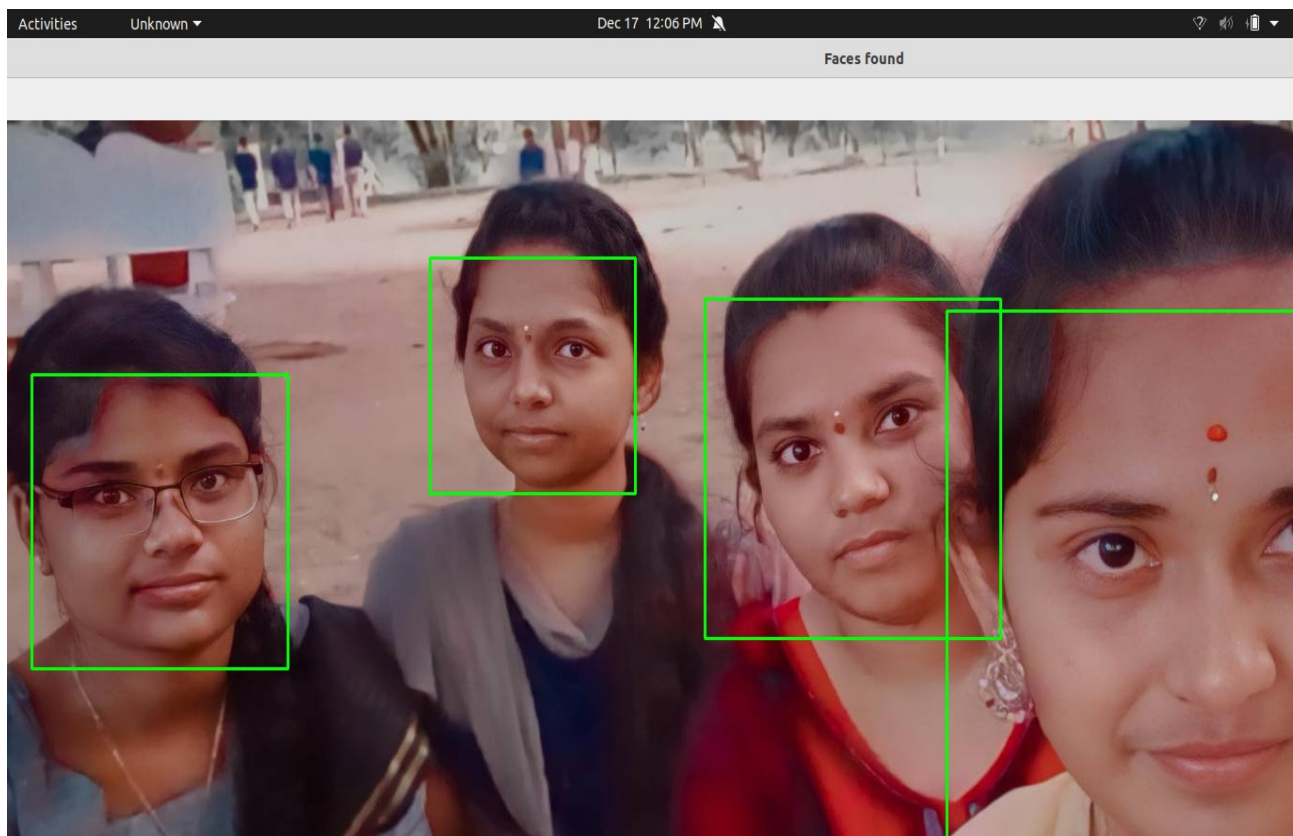


Fig.2

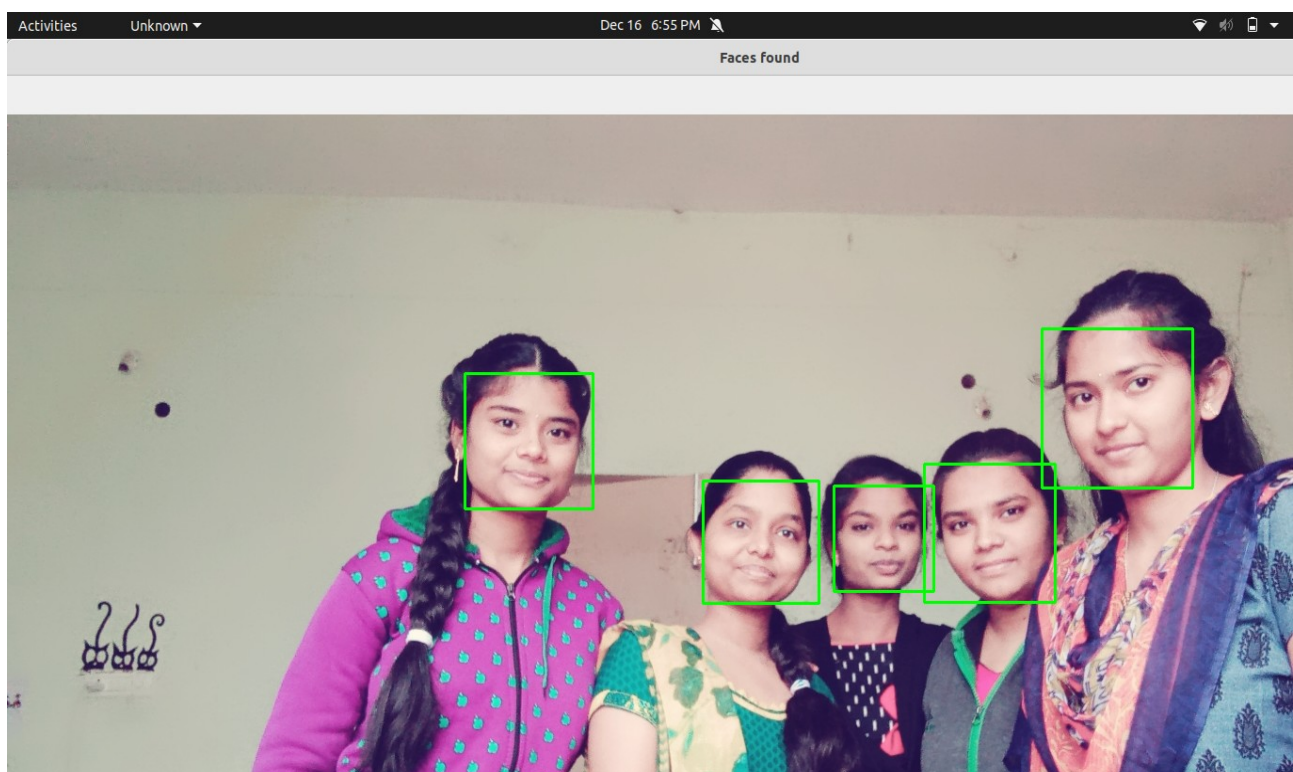


Fig.3



webcam:



Fig.4

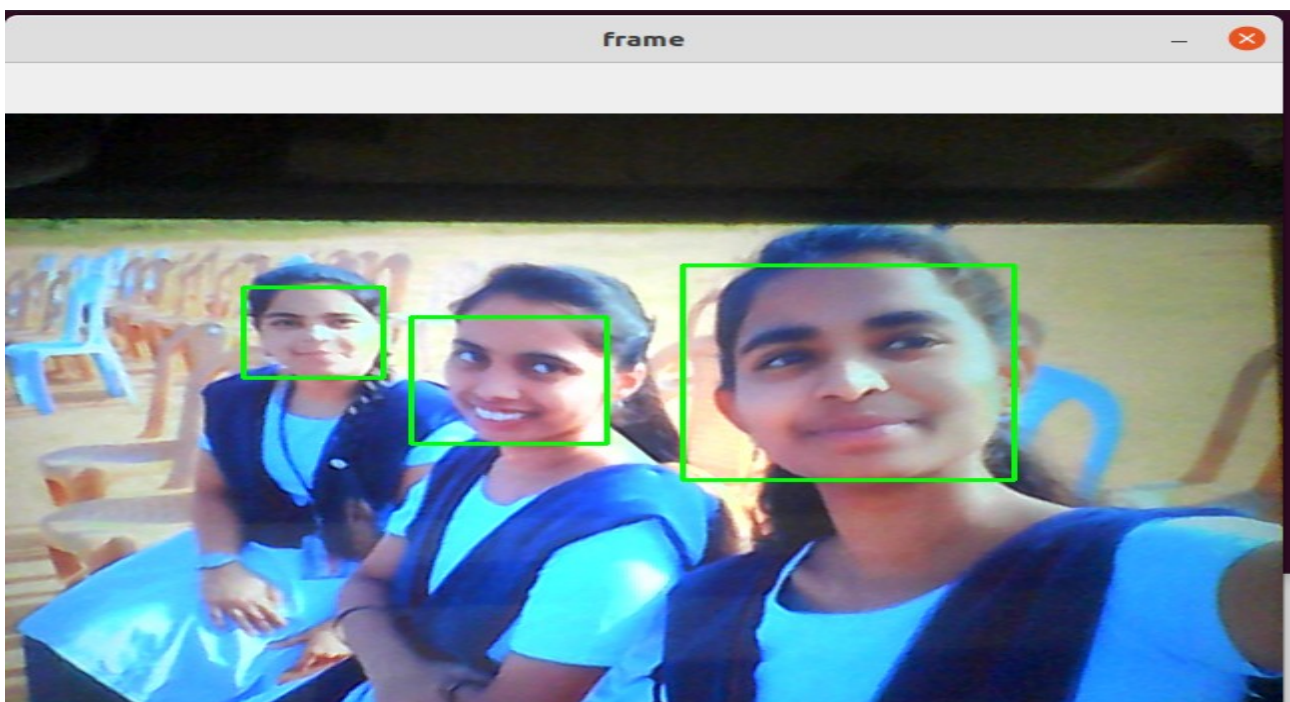
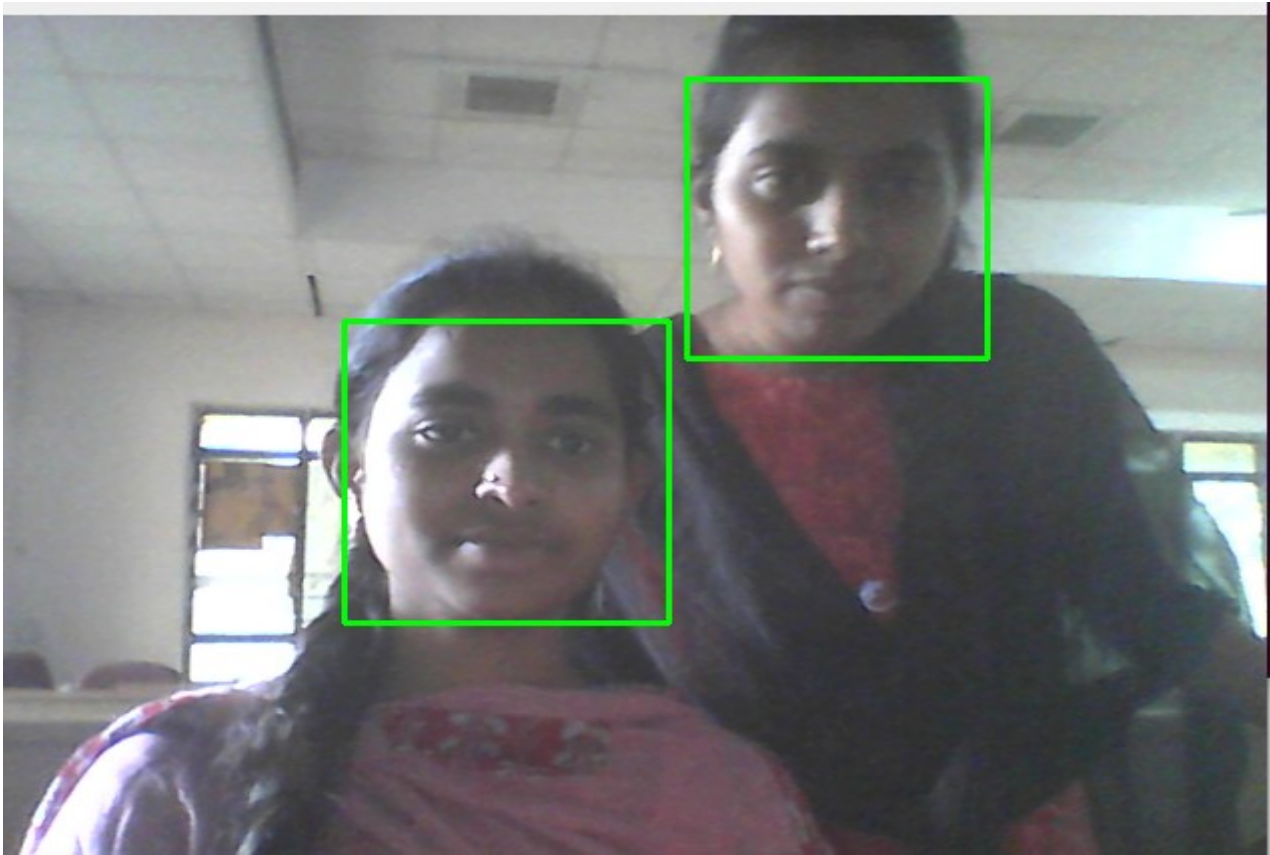


Fig.5



**Fig.6**

### **Scope of Project:**

- ✓ To identify and verify terrorists at airports, railway stations and malls the face recognition technology will be the best
- ✓ choice in India as compared with other biometric technologies since other technologies cannot be helpful in crowded places.
- ✓ It can also be deployed in police station to identify and verify the criminals.
- ✓ To Identify the Number of student in classroom. It can collect in all the number of faces in less than 1 sec. Easy to take attendance.

# CHAPTER- 4

## 4.1 CONCLUSION:

Face recognition technologies have been associated generally with very costly top secure Applications. Today the core technologies have evolved and the cost of equipments is going down dramatically due to the integration and the increasing processing power. Certain Applications of face recognition technology are now cost effective, reliable and highly Accurate.

Government and ngos should concentrate and promote applications of facial recognition System in india in various fields by giving economical support and appreciation.

A variety of possible advantages come with the usage of facial recognition. There is no need to directly touch an authentication system relative to other touch-based biometric identification methods such as fingerprint scanners, which could not function well if a person's hand is soil.

- The safety standard changed.
- It needs less preparation than other methods for biometric authentication.
- Fast to integrate with current protection applications.
- Through time the precision of the measurements has increased.
- It could be used to help in the authentication workflow.

## 4.2 Future works:

Face detection may be used for a broad range of purposes, from defense to ads. Any examples in usage include Smartphone makers, including Apple, for public protection.

- **Government** at airports, by the Homeland Security Agency, to recognize people who can meet their visa criteria.
- **Law enforcement** by gathering mugshots can evaluate local, national, and federal assets repositories too.
- 
- **Social networking** is used for identifying individuals in photos, which also includes Twitter.
- **Business protection**, as businesses may use facial recognition to access their buildings.
- **Marketing** ,where advertisers may use facial recognition to assess particular age, gender, and ethnicity

# CHAPTER - 5

## References:

<https://www.datacamp.com/tutorial/face-detection-python-opencv>

<https://github.com/Vatshayan/Face-Detection-Project>

<https://github.com/Akkmr111/Face-recognition-using-opencv-and-python>

<https://pyimagesearch.com/2021/04/05/opencv-face-detection-with-haar-cascades/>