Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

Student Name

***Sanatkumar Rajmogali Ippalpalli***

Title of Project Report

***Guided Project 17 – Face Detection***

Table of Contents

[Introduction 4](#_Toc84256729)

[Eckovation theme & Question 5](#_Toc84256730)

[Prerequisites before starting coding 6](#_Toc84256731)

[program DEVELOPMENT steps 8](#_Toc84256732)

[Dataset requirements 8](#_Toc84256733)

[Technique – OpenCV 8](#_Toc84256734)

[PROGRAM / CODE DEVELOPMENT 9](#_Toc84256735)

[Analysis 12](#_Toc84256736)

[CONCLUSION 13](#_Toc84256737)

[Figure 1 Import libraries and datasets/modules 9](#_Toc84256722)

[Figure 2 Load Image 9](#_Toc84256723)

[Figure 3 Cascade Classifier 9](#_Toc84256724)

[Figure 4 Read Image and convert to gray 9](#_Toc84256725)

[Figure 5 Code for plotting images 10](#_Toc84256726)

[Figure 6 results for plotting images 10](#_Toc84256727)

[Figure 7 Plotting of Images with face detected 11](#_Toc84256728)

***EXECUTIVE SUMMARY***

Science and technology improved many technologies and has guided numerous innovative features which advanced the smart phones and laptops and smart watches that we use daily. Security features designs are meant to protect personal information and accurate biometric recognition. The biometric recognition innovative technology included user defined (4/6/8 digit pins/passwords/patterns), fingerprint sensors and facial recognition. *One of the latest feature is facial recognition is secure, safe and one of the enhanced, best biometric identification process to identify, verify, and authenticate the person using facial features from any photo or video or live to unlock the device.*

As part of guided project, one of the feature is the facial recognition to detect face which is coded & implemented to capture the details as mentioned in the question.

Face detection is a computer technology being used in a variety of applications that identifies human faces in digital images. Face detection can be regarded as a specific case of object-class detection, which focuses on the detection of frontal human faces. Because faces are so complicated, there isn’t a straightforward test that will tell if we found a face or not. Instead, thousands of small patterns and features must be matched for accurate facial detection.

# Introduction

Science and technology improved many technologies and has guided numerous innovative features which advanced the smart phones and laptops and smart watches that we use daily. Security features designs are meant to protect personal information and accurate biometric recognition. The biometric recognition innovative technology included user defined (4/6/8 digit pins/passwords/patterns), fingerprint sensors and facial recognition. *One of the latest feature is facial recognition is secure, safe and one of the enhanced, best biometric identification process to identify, verify, and authenticate the person using facial features from any photo or video or live to unlock the device.*

Face detection is important part of surveillance systems and it has been widely used in computer vision and image processing. Face detection is also first step of the facial feature extraction. Facial feature extraction is a topic that has been focused on by many researchers in computer science, psychology, medicine and related fields and has become increasingly important in recent years. With the help of facial features, machine learning algorithms can estimate ages and classify genders of people.

Facial feature extraction consists in localizing the most characteristic face components (eyes, nose, mouth, etc.) within images that depict human faces. This step is essential for the initialization of many face processing techniques like face tracking, facial expression recognition or face recognition. Among these, face recognition is a lively research area where it has been made a great effort in the last years to design and compare different techniques.

Hence Eckovation includes this guided project in the courseware for students to understand, implementation / execute the code themselves.

This report includes the 5W1H about the theme of development of code and running the code with database available over the internet. At the end of the report, the conclusions share the features extracted and useful for next course of activities to gain advantages in the facial recognition activities development.

# Eckovation theme & Question

**Theme : Face Detection**

Face detection is a computer technology being used in a variety of applications that identifies human faces in digital images. Face detection can be regarded as a specific case of object-class detection, which focuses on the detection of frontal human faces. Because faces are so complicated, there isn’t a straightforward test that will tell if we found a face or not. Instead, thousands of small patterns and features must be matched for accurate facial detection.

**Question:**

Implement a face tracking algorithm using haar cascade algorithm and opencv. Using haar cascade, first implement a face detection algorithm that counts the total number of faces present in any given frame. Write the total number of faces detected on the top left of the image. Further modify the code to track the face if only one face is detected. Make sure that you draw the bounding box corresponding to all video frames. Note: you may need to fine tune the parameters for Haar Cascade Classifier to get optimal results and remove false positives.

# Prerequisites before starting coding

1. Who - Software needed?
2. What - Version / Release of software?
3. Any Prerequisites
4. How - to install the software
5. Which -libraries are needed to execute the problem statement
6. Where – dataset requirements, path location to include in the code
7. When – to use the above feature extraction
8. Who – Software neeed?

Python

1. What- Version / Release of software?

Python version 3.6 (latest version of python)

1. Any Prerequisites

RAM space availability & hard disk space availability

Admin rights to install the software

1. How - to install the software
2. The following url <https://www.python.org/downloads/>can be referred to download python.
3. Second and easier option is to download anaconda and use its anaconda prompt to run the commands. To install anaconda check this url <https://www.anaconda.com/download/>
4. Which -libraries are needed to execute the problem statement
5. OpenCv as cv2
6. Matplotlib (pip install matplotlib)
7. sys
8. Where – dataset requirements, path location to include in the code
9. Once you have python downloaded and installed, you will need to setup PATH variables (if you want to run python program directly, detail instructions are below in how to run software section). To do that check this: [https://www.pythoncentral.io/add-python-to-path-python-is-not- recognized-as-an-internal-or-external-](https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/) [command/](https://www.pythoncentral.io/add-python-to-path-python-is-not-recognized-as-an-internal-or-external-command/).
10. Setting up PATH variable is optional as you can also run program without it and more instruction are given below on this topic.
11. When – to use the above feature extraction
12. When – to use the above face detection

# program DEVELOPMENT steps

* Dataset requirement
* Technique selections
* Program / code development
* Analysis

### Dataset requirements

The data set contains 2 images of faces collected from the web.

### Technique – OpenCV

**OpenCV** library in python is a computer vision library that is widely used for image analysis, image processing, detection, recognition, etc.

Let us hop to the inscribing carving!

### PROGRAM / CODE DEVELOPMENT

As explained step by step during the lecture by mentor, we would approach steps and understand the basics with brief explanation as needed.

#### Step 1: Import the relevant libraries and applicable datasets/modules

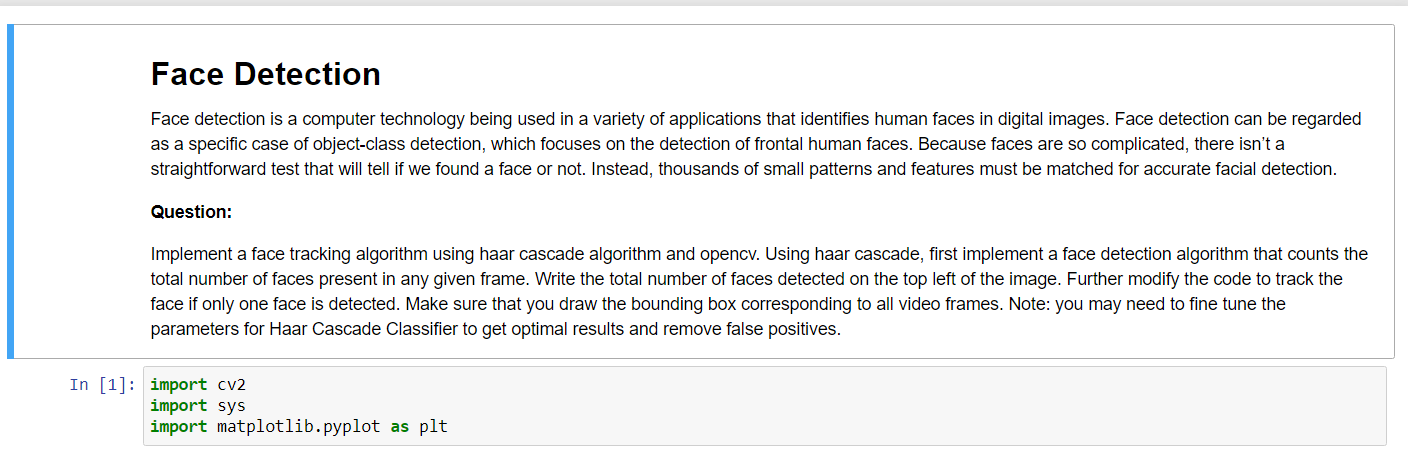


Figure 1 Import libraries and datasets/modules

#### Step 2: Load Images &Haar Cascade

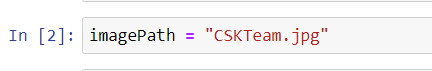


Figure 2 Load Image

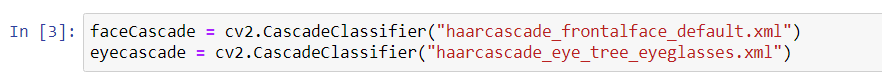


Figure 3 Cascade Classifier

Graphical user interface, text, application

Description automatically generated

Figure Read Image and convert to gray

#### Step 3: Plot the images

Loop over images to plot the images.

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated

Figure Code for plotting images

Graphical user interface, text

Description automatically generated

Figure 6 results for plotting images

#### Step 4: Full program/code in one go implementation

Text

Description automatically generated

A picture containing text, screen

Description automatically generated

Figure 7 Plotting of Images with face detected

### Analysis

Using Haar Cascade XML files and OpenCV, we are able to detect faces easily.

This entire program runs within 5 seconds.

# CONCLUSION

In this guided project, we built a code for detection of face using HAAR CASCADE & OPENCV.

Hence, the improvements in the code with time with multiple attempts may be checked and justified for the accuracy score.

This entire program runs within few seconds.

references:

<https://www.geeksforgeeks.org/face-detection-using-python-and-opencv-with-webcam/>

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