

Project Report on
Face Recognize

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

SAMEER VERMA (21SCSE1011328)



**SCHOOL OF COMPUTING SCIENCE
AND ENGINEERING,
GREATER NOIDA, UTTAR PRADESH**

MAY 2023

INDEX

S. NO	TOPIC	PAGE NO
1	Chapter 1:- Introduction 1.1. Background 1.2. Methodology 1.3. Implementation 1.4. OpenCV	
2	Chapter 2:- Source Code	
3	Chapter 3:- Output , Results and Discussion	
4	Chapter 4:- Conclusion	
5	Chapter 5:- Reference	

Chapter 1

Introduction:

The purpose of this report is to present an implementation of real-time face detection using OpenCV, a popular computer vision library. The report will discuss the code implementation and provide an overview of the underlying concepts and techniques used for face detection.

Emotion Analysis: Facial expressions provide insights into human emotions. By detecting faces and analysing their expressions, computer vision systems can infer emotions, which has applications in areas like market research, psychology, and human-computer interaction.

Human-Computer Interaction: Face detection allows computers to interact with users in a more intuitive and natural way. It enables systems to track head movements, detect eye gaze, and recognize facial gestures, leading to advancements in augmented reality, virtual reality, and user interface design.

1.1. Background:

- Brief introduction to face detection and its importance in computer vision applications.
- Overview of OpenCV and its capabilities for image and video processing.

1.2. Methodology:

- Description of the code structure and its various components.
- Explanation of the Haar Cascade classifier and its role in face detection.
- Discussion of the parameters used for face detection, such as scale Factor, minNeighbors, and minSize.
- Overview of the process for converting frames to grayscale and detecting faces.

1.3. Implementation:

- Step-by-step explanation of the code segments.
- Demonstration of how the pre-trained face detection model is loaded.
- Explanation of the video capture initialization and frame retrieval process.
- Discussion of the face detection algorithm and the drawing of rectangles around detected faces.
- Description of how the resulting frames are displayed in real-time.

1.4. OpenCV:

OpenCV, short for Open Source Computer Vision Library, is an open-source computer vision and machine learning software library. It provides a comprehensive set of tools, functions, and algorithms to facilitate various tasks related to image and video processing, object detection and tracking, machine learning, and more. OpenCV was initially developed by Intel in 1999 and has since been supported by a large community of developers worldwide

Chapter 2

Source Code:

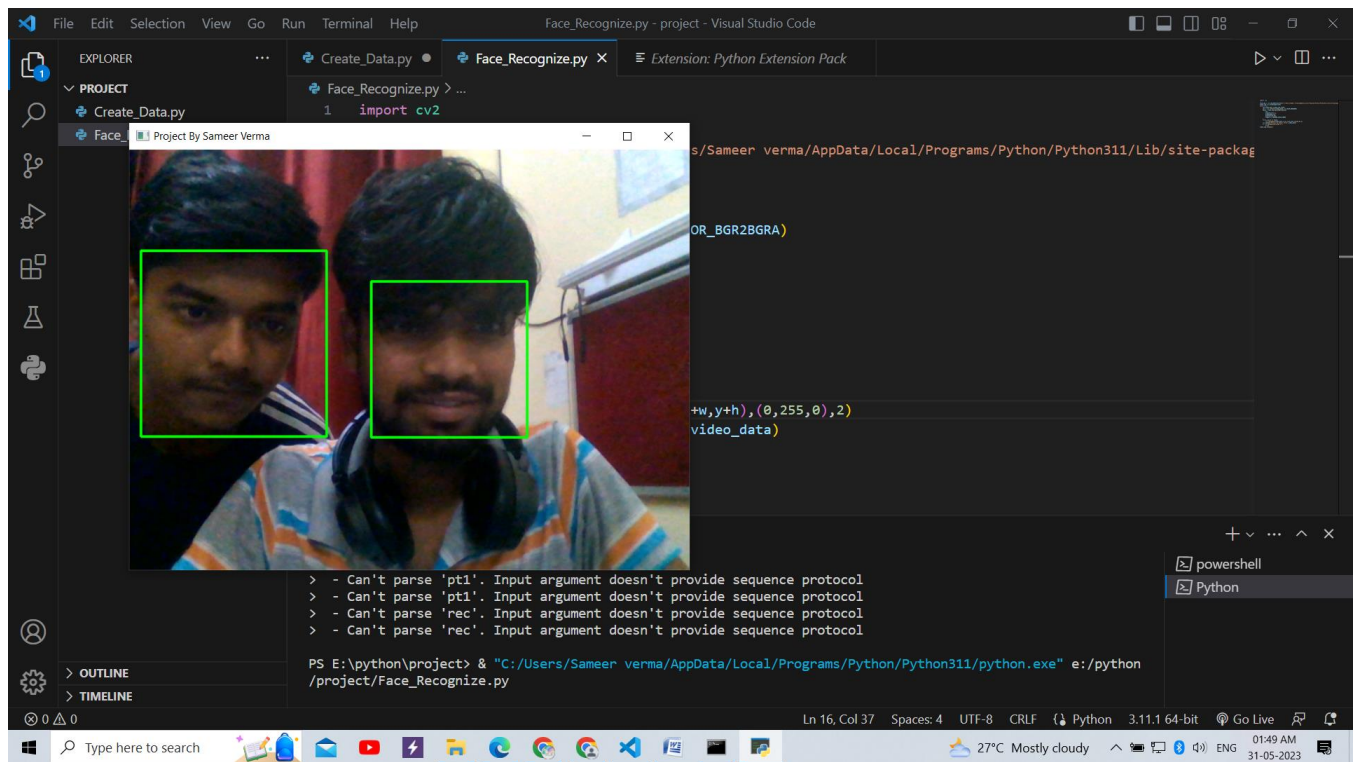
```
import cv2

face_cap = cv2.CascadeClassifier("C:/Users/Sameer
verma/AppData/Local/Programs/Python/Python311/Lib/site-
packages/cv2/data/haarcascade_frontalface_default.xml")
video_cap =cv2.VideoCapture(0)
while True:
    ret,video_data =video_cap.read()
    col = cv2.cvtColor(video_data,cv2.COLOR_BGR2BGRA)
    faces = face_cap.detectMultiScale(
        col,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30,30),
        flags=cv2.CASCADE_SCALE_IMAGE
    )
    for(x,y,w,h) in faces:
        cv2.rectangle(video_data,(x,y),(x+w,y+h),(0,255,0),2)
    cv2.imshow("Project By Sameer Verma",video_data)
    if cv2.waitKey(10)==ord('a'):
        break
video_cap.release()
```

Chapter 3

Output, Results and Discussion

3.1 Output



3.2. Results and Discussion:

- Presentation of the results obtained from running the code.
- Evaluation of the face detection performance in different scenarios.
- Analysis of the strengths and limitations of the implemented approach.

Chapter 4

Conclusion:

- Recap of the main points discussed in the report.
- Summary of the benefits and applications of real-time face detection using OpenCV.
- Future possibilities for improvement or expansion of the implemented code.
- Appendix (Code Listing):
 - Full code listing of the face detection implementation using OpenCV.
 - Instructions for installing the required packages and running the code.

Chapter 5

References:

- List of sources used for background information and research.
- Note: The report structure provided above is a suggested outline. You can modify or expand it based on your specific requirements and guidelines. Additionally, don't forget to include appropriate citations and references for any external sources used in the report.