Emotion Detection by Applying CNN

Varun Jagadeeshwar Jonnalagadda [1], Sri Santhan Josyula [2]

[1, 2] Graduate Engineer,
Department of Computer Science Engineering, Sri Chandra Sekharendra Sarawathi Viswa Maha Vidyalaya (SCSVMV) University,
Kanchipuram, Tamil Nadu, India.

Published in IJIRMPS (E-ISSN: 2349-7300), Volume 11, Issue 5, (September-October 2023)

Abstract
As the human-robot interaction is catching the eye day by day with the increase in need for automation in every field, personal robots are increasing in every area which may be the coping needs of elderly people, treating autistic patients, or child therapy, even in the area of babysitting the child.

In this project, we are going to build a model which can predict human emotion from an image in real-time on a module based on web design where we can explore user's emotions through surveys.

Keywords: Artificial Intelligence, Machine Learning, Deep Learning, Convolutional Neural Network, JavaScript, FER 2013, Facial Expressions, Python, Kaggle, HTML, Web Design

Introduction

Emotion Detection using CNN (Machine Learning)

This project can

1. Extract and leverage the facial expressions
2. Gather the information of the facial language traits (i.e., facial posture, facial gestures/movements etc.)

The detection and observation of emotions are important in various situations i.e., to enable naturalistic and personalized human-computer interaction.

Emotion detection often requires modeling of various data inputs such as facial expressions, e.g., anger, sadness, happiness, etc.

From data collected by this technology, provides a unique opportunity for some unimaginable ways in recognising human activities through human physiological sensing.

This could potentially provide better management of chronic diseases such as asthma, diabetes and cardiovascular diseases

Emotion recognition can also be useful in variety of areas such as image processing, cyber security, robotics, psychological studies, and virtual reality applications.

Objectives

- Project objectives are what you plan to achieve by the end of your project.
- This might include deliverables and assets, or more intangible objectives like increasing productivity or motivation.
- By the end of this project we are going to visualize the emotions of the user through a website module
- For which we are using software like Kaggle, Keras, etc. programming languages like python,

JavaScript, etc. markup languages like HTML and also datasets (FER 2013).
- By using our project we can process images and videos in real-time for monitoring video feeds or automating videos analytics which saves money and makes a better life for necessities.

Scope of the Project

- The project we are doing is a web design model where user can know their state of mind in a survey mode through a web module.
- In which they are going to select seven random emotions in the website where they act like emotion triggers capturing the facial expressions of the user.
- With a survey module we will show the user their emotional results.
- Whether the user is a sociopath or psychopath by explaining the percentage through an emotion detector.

Literature Survey

Title	**Authors**	**Year of Publication**
An efficient deep learning technique for facial emotion recognition	Asad Khattak, Muhammad Zubair Asghar, Mushtaq Ali, Ulfat Batool	2022
Facial Feature-Based Human Emotion Detection Using Machine Learning	Mritunjay Rai, Agha Asim Husain, Rohit Sharma, Tanmoy Maity, R.K. Yadav	2022
Automated emotion recognition: Current trends and future perspectives	M. Maithri, U. Raghavendra, Anjan Gudigar, Jyothi Samanth, Prabal Datta Barua, Murugappan Murugappan, Yashas Chakole, U. Rajendra Acharya	2022

Tools Used

- Artificial Intelligence
- Machine Learning
- Deep Learning
- Convolutional neural networks
- JavaScript
- FER 2013
- Facial expressions
- Python
- Kaggle
- HTML
- Web design

Problem Statement

Problem

The main purpose of this project is to know the emotion of the user where it develops the human-computer interaction.

Problem Statement

- As robots are helping human beings in most of the cases, robots need to understand human emotion in order to treat human in a more customized manner.
- Predicting human emotion has been a difficult problem for robots which hadn’t been solved over a decade’s time.

- We are solving that problem by using convolutional neural networking's in which we can detect human emotion for greater use.

Objectives

- For which we are using software like Kaggle, Keras, etc. programming languages like python, JavaScript, etc. markup languages like HTML and also datasets (FER 2013).
- By using our project we can process images and videos in real-time for monitoring video feeds or automating videos analytics which saves money and makes a better life for necessities.

Methodology

This project helps to entertain the user in expressive way where they can find their own emotion.

Proposed Methodology

As per the main methodology used in this project, we need 2 modules:

1. Emotion detector, which works on the back-end.
2. Web design module, which uses emotion detector.

For that, we need some libraries, softwares, languages, etc.

First, in the emotion detection, we use a dataset called FER (Facial Expression Recognition) to analyze the facial expressions (i.e., emotion-detection.py) for classifying the emotion with CNN using Keras on the Kaggle. After completion of the code, it will give us an h5 file to download. Later we use that file on the main python code as a dataset directory on PyCharm. We also use HAAR cascade frontal-face for the Object Detection Algorithm in the main code (main.py). After giving the directories and attaching the dataset to the (main.py) code it will display the result by showing the emotions by using a webcam

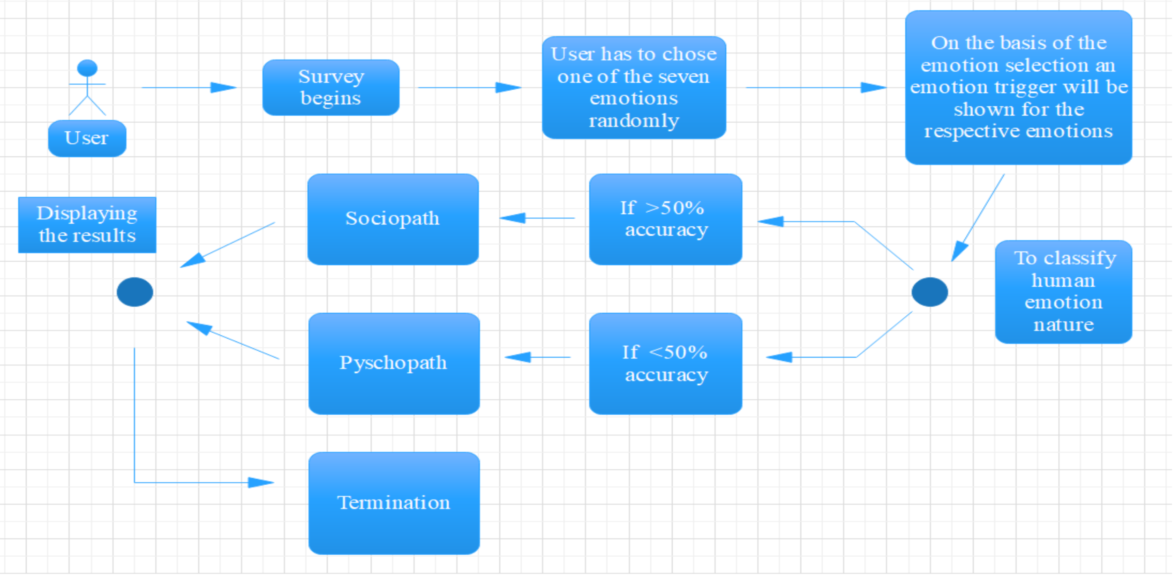
From there we build a web application by using the templates folder containing HTML files and web design, for implementing our second module. For that we need a library called Flask which helps you to provide tools and technologies that allow you to build a web application in python. By using flask, we write a code with i.e., app.py which also contains files to build a web app in python which uses emotion detection files from the module to run in the web application. By this, our second module will be completed and that is how we implement our project.

Architecture

The main objective of this system architecture is to detect the emotion so the approach of that process goes like this: AI - ML - Deep Learning - CNN - Data Sets - Kaggle - Python - Web Design - Facial Recognition.

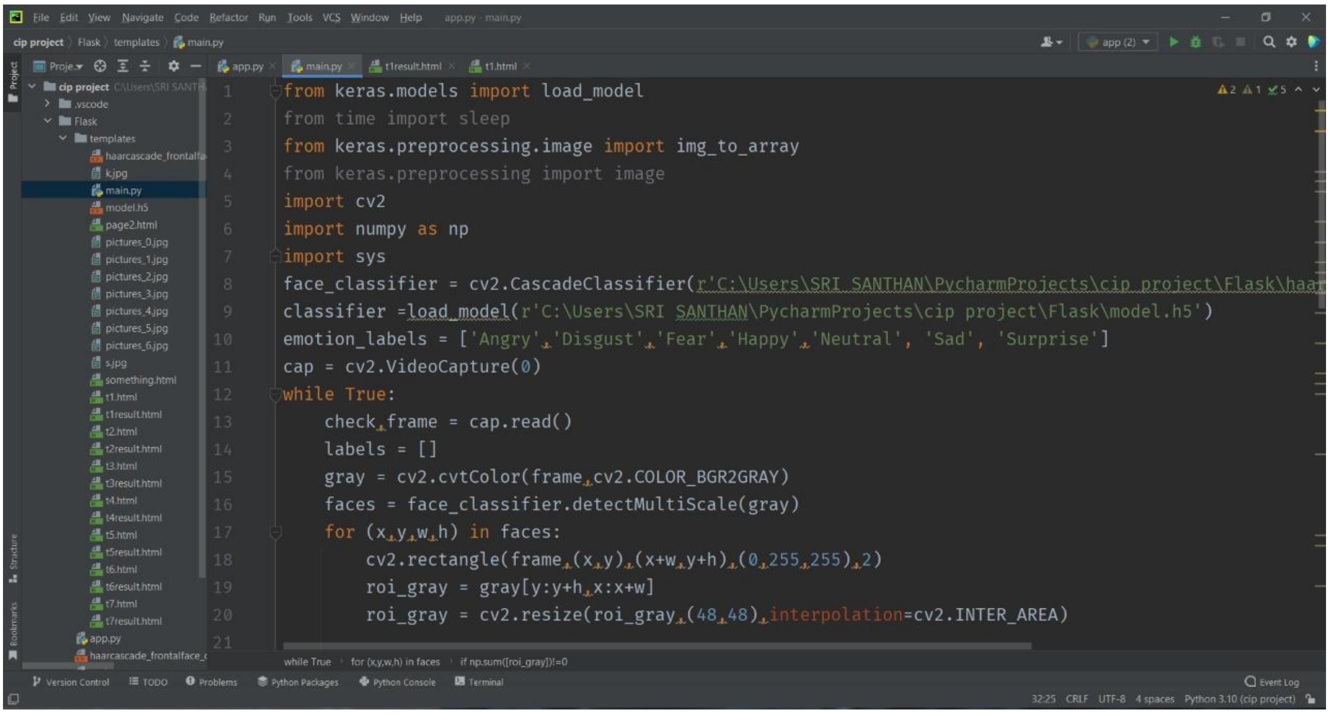
As can be seen, there are 9 stages to approach this project.

UML Diagram

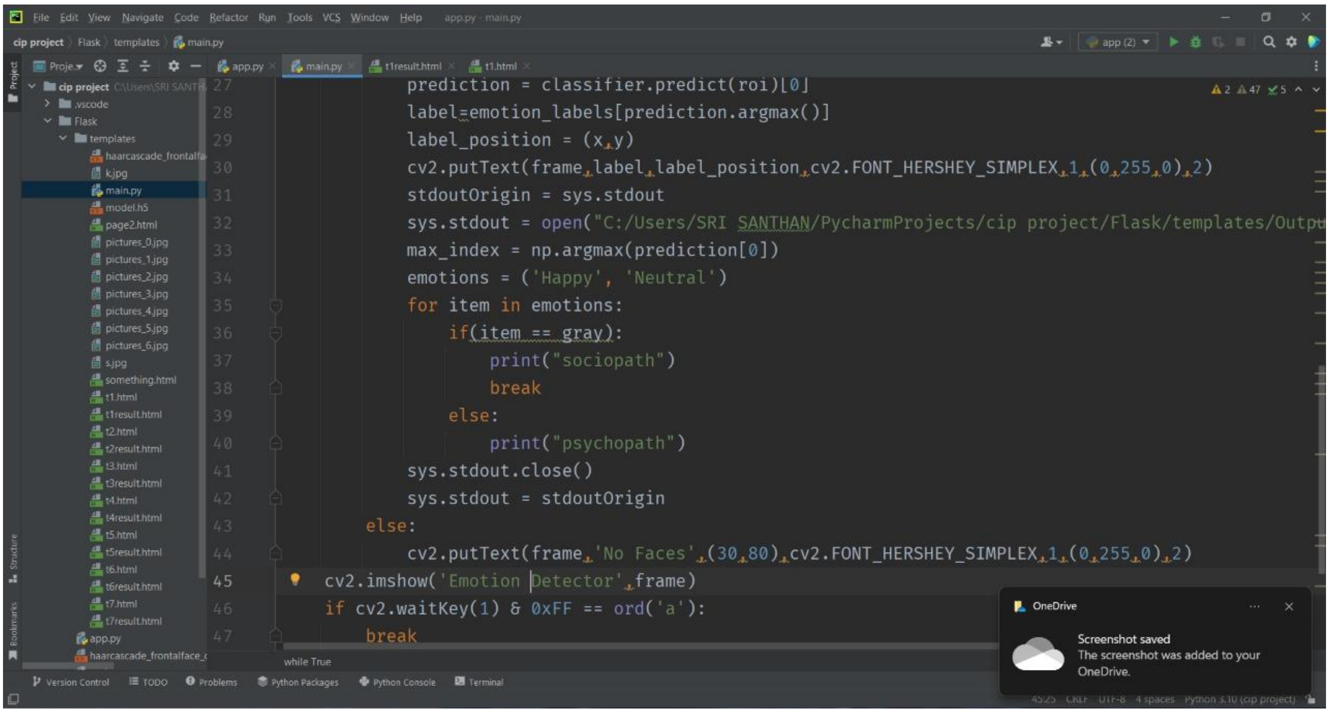

User
Survey begins
User has to chose one of the seven emotions randomly
On the basis of the emotion selection an emotion trigger will be shown for the respective emotions
To classify human emotion nature
If >50% accuracy
If <50% accuracy
Sociopath
Pyschopath
Displaying the results
Termination

Result

Sample Coding

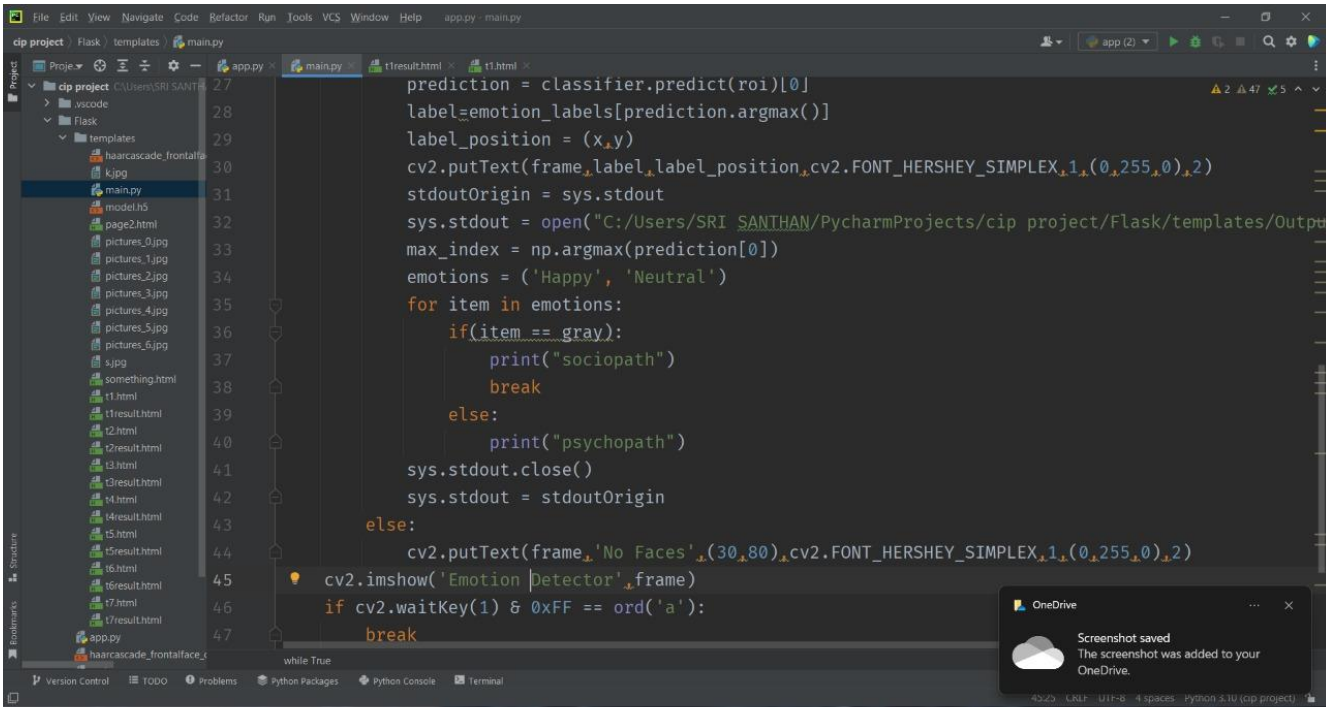


```python
from keras.models import load_model
from time import sleep
from keras.preprocessing.image import img_to_array
from keras.preprocessing import image
import cv2
import numpy as np
import sys
face_classifier = cv2.CascadeClassifier(r'C:\Users\SRI SANTHAN\PycharmProjects\cip project\Flask\haa
classifier =load_model(r'C:\Users\SRI SANTHAN\PycharmProjects\cip project\Flask\model.h5')
emotion_labels = ['Angry','Disgust','Fear','Happy','Neutral', 'Sad', 'Surprise']
cap = cv2.VideoCapture(0)
while True:
    check,frame = cap.read()
    labels = []
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray)
    for (x,y,w,h) in faces:
        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,255),2)
        roi_gray = gray[y:y+h,x:x+w]
        roi_gray = cv2.resize(roi_gray,(48,48),interpolation=cv2.INTER_AREA)
```



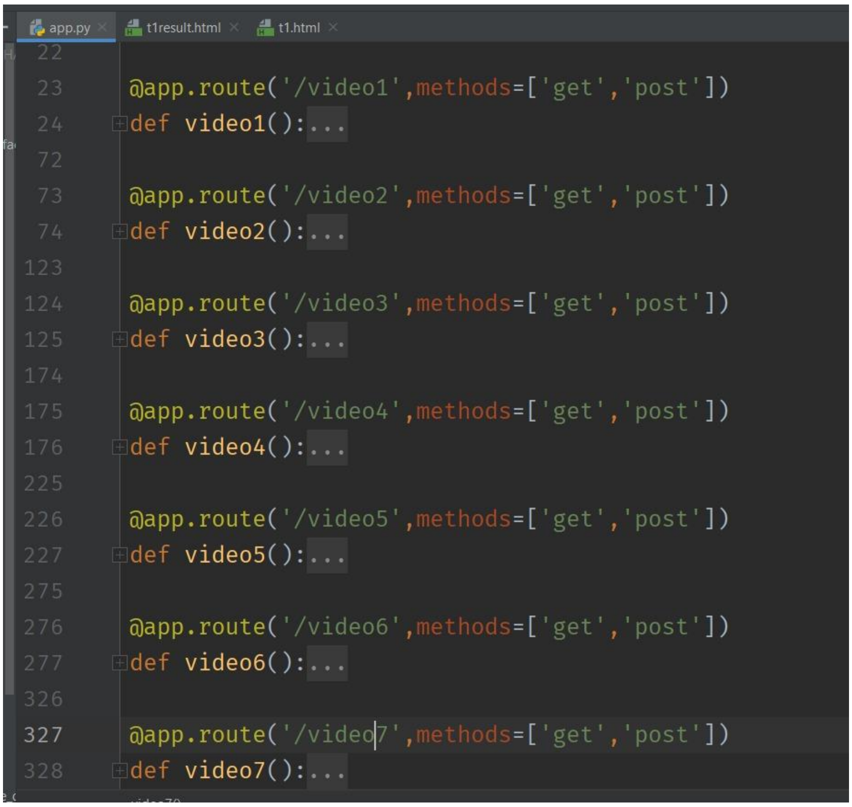
```
            prediction = classifier.predict(roi)[0]
            label=emotion_labels[prediction.argmax()]
            label_position = (x,y)
            cv2.putText(frame,label,label_position,cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)
            stdoutOrigin = sys.stdout
            sys.stdout = open("C:/Users/SRI SANTHAN/PycharmProjects/cip project/Flask/templates/Outpu
            max_index = np.argmax(prediction[0])
            emotions = ('Happy', 'Neutral')
            for item in emotions:
                if(item == gray):
                    print("sociopath")
                    break
                else:
                    print("psychopath")
            sys.stdout.close()
            sys.stdout = stdoutOrigin
        else:
            cv2.putText(frame,'No Faces',(30,80),cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)
    cv2.imshow('Emotion Detector',frame)
    if cv2.waitKey(1) & 0xFF == ord('a'):
        break
```

OneDrive
Screenshot saved
The screenshot was added to your OneDrive.

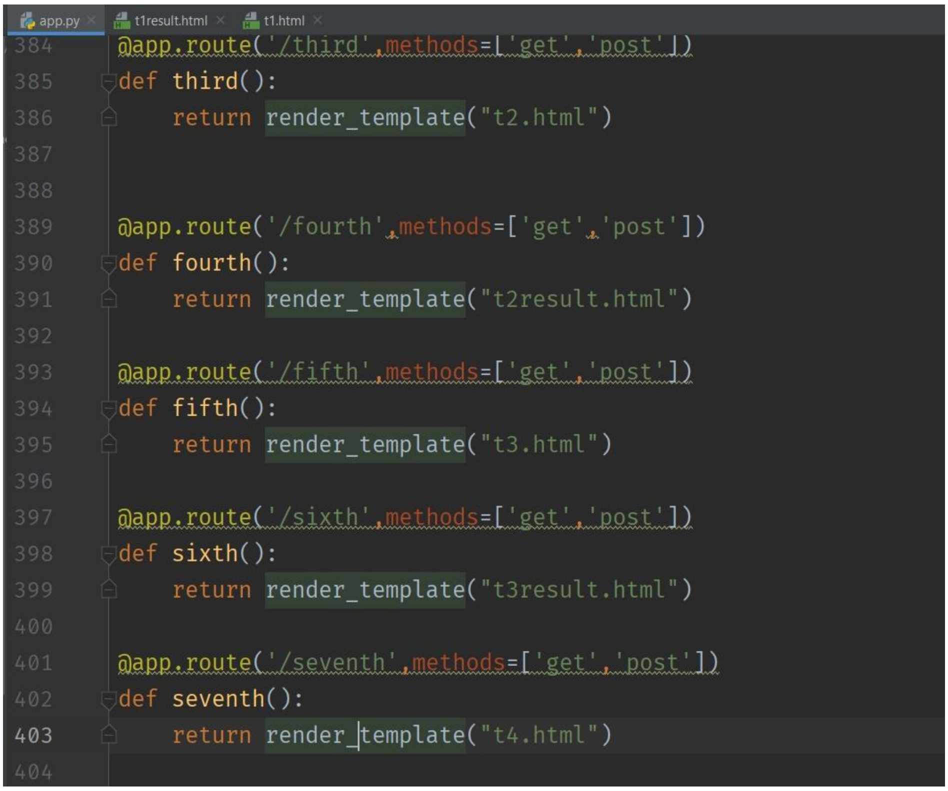

cip project 〉 Flask 〉 templates 〉 main.py

```
            prediction = classifier.predict(roi)[0]
            label=emotion_labels[prediction.argmax()]
            label_position = (x,y)
            cv2.putText(frame,label,label_position,cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)
            stdoutOrigin = sys.stdout
            sys.stdout = open("C:/Users/SRI SANTHAN/PycharmProjects/cip project/Flask/templates/Outpu
            max_index = np.argmax(prediction[0])
            emotions = ('Happy', 'Neutral')
            for item in emotions:
                if(item == gray):
                    print("sociopath")
                    break
                else:
                    print("psychopath")
            sys.stdout.close()
            sys.stdout = stdoutOrigin
        else:
            cv2.putText(frame,'No Faces',(30,80),cv2.FONT_HERSHEY_SIMPLEX,1,(0,255,0),2)
    cv2.imshow('Emotion Detector',frame)
    if cv2.waitKey(1) & 0xFF == ord('a'):
        break
```

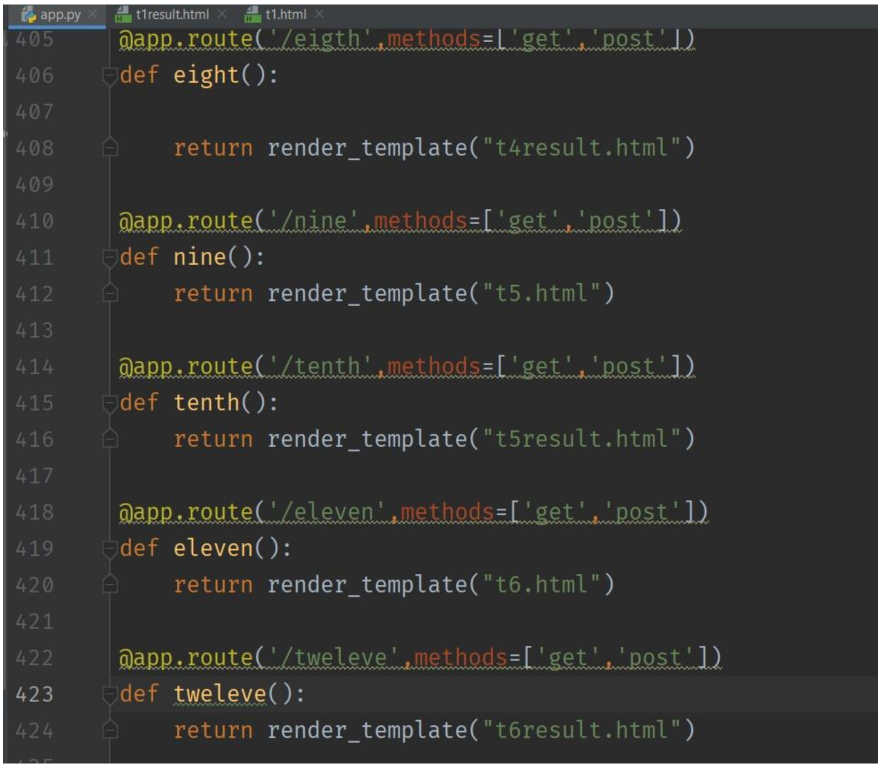
OneDrive
Screenshot saved
The screenshot was added to your OneDrive.



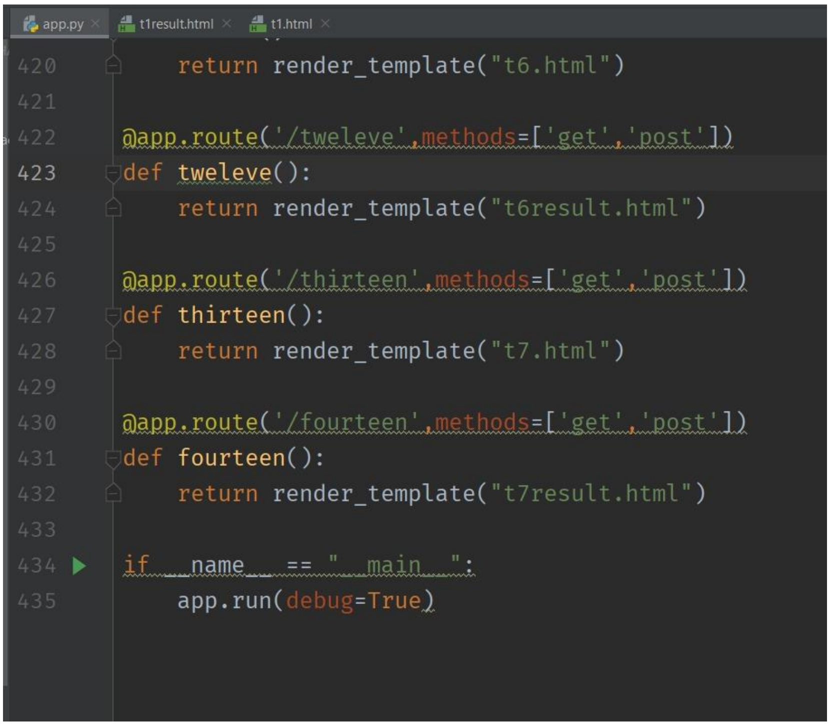
```

@app.route('/video1',methods=['get','post'])
def video1():...

@app.route('/video2',methods=['get','post'])
def video2():...

@app.route('/video3',methods=['get','post'])
def video3():...

@app.route('/video4',methods=['get','post'])
def video4():...

@app.route('/video5',methods=['get','post'])
def video5():...

@app.route('/video6',methods=['get','post'])
def video6():...

@app.route('/video7',methods=['get','post'])
def video7():...
```



```
@app.route('/third',methods=['get','post'])
def third():
    return render_template("t2.html")


@app.route('/fourth',methods=['get','post'])
def fourth():
    return render_template("t2result.html")

@app.route('/fifth',methods=['get','post'])
def fifth():
    return render_template("t3.html")

@app.route('/sixth',methods=['get','post'])
def sixth():
    return render_template("t3result.html")

@app.route('/seventh',methods=['get','post'])
def seventh():
    return render_template("t4.html")

```

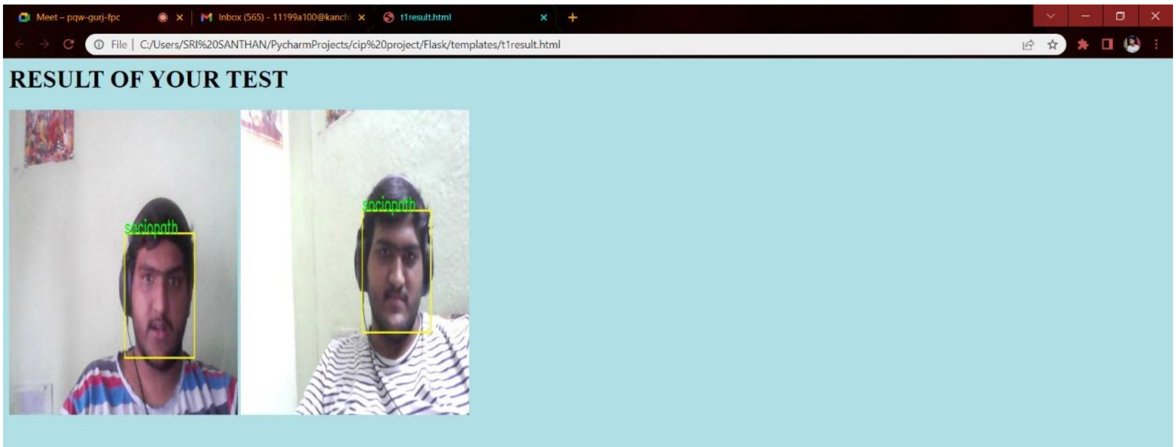


```python
@app.route('/eigth',methods=['get','post'])
def eight():

    return render_template("t4result.html")

@app.route('/nine',methods=['get','post'])
def nine():
    return render_template("t5.html")

@app.route('/tenth',methods=['get','post'])
def tenth():
    return render_template("t5result.html")

@app.route('/eleven',methods=['get','post'])
def eleven():
    return render_template("t6.html")

@app.route('/tweleve',methods=['get','post'])
def tweleve():
    return render_template("t6result.html")
```



```python
    return render_template("t6.html")

@app.route('/tweleve',methods=['get','post'])
def tweleve():
    return render_template("t6result.html")

@app.route('/thirteen',methods=['get','post'])
def thirteen():
    return render_template("t7.html")

@app.route('/fourteen',methods=['get','post'])
def fourteen():
    return render_template("t7result.html")

if __name__ == "__main__":
    app.run(debug=True)
```

Sample Output


Meet – pqw-gurj-fpc
Inbox (565) - 11199a100@kanch
t1result.html
File | C:/Users/SRI%20SANTHAN/PycharmProjects/cip%20project/Flask/templates/t1result.html

RESULT OF YOUR TEST

This is how it is going to show the result.

Conclusion

- Emotion detection is a great technology, which develops human-computer interaction, in many ways.
- Emotion recognition can also be useful in a variety of areas such as image processing, cyber security, robotics, psychological studies, and virtual reality applications.
- From data collected by this technology, provides a unique opportunity for some unimaginable ways in recognising human activities through human physiological sensing.

Hence, this is our project on emotion detection.

References

Emotion Detection

[1] Eiman Kanjo, Eman M.G. Younis, Chee Siang Ang, "Deep learning analysis of mobile physiological, environmental and location sensor data for emotion detection", Information Fusion, 49, 2019, 46-56

[2] Kashfia Sailunaz, Manmeet Dhaliwal, Jon Rokne, Reda Alhajj, "Emotion detection from text and speech: A survey".

[3] Shruti Jaiswal, G.C. Nandi, "Robust real-time emotion detection system using CNN architecture".

[4] E. Mower Provost, "Identifying salient sub-utterance emotion dynamics using flexible units and estimates of affective flow", in Proceedings of IEEE ICASSP 2013.

[5] Nitisha Raut, "Facial Emotion Recognition Using Machine Learning", San Jose State University.

[6] Luz Santamaria-Granados, Mario Munoz-Organero, Gustavo Ramirez-González, Enas Abdulhay, N. Arunkumar, "Using Deep Convolutional Neural Network for Emotion Detection on a Physiological Signals Dataset (AMIGOS)".

[7] W. Swinkels, L. Claesen, F. Xiao, H. Shen, "SVM point-based real-time emotion detection", 2017 IEEE Conference on Dependable and Secure Computing, Taipei.

[8] Daniela Girardi, Filippo Lanubile, Nicole Novielli, "Emotion detection using noninvasive low cost sensors", University of Bari 'Aldo Moro'.

Web Development

[9] J. Frocier, P. Bissex, W.J. Chun, "Python web development with Django", 2008.

[10] K. Lei, Y. Ma, Z. Tan, "Performance comparison and evaluation of web development technologies in PHP, Python, and Node.js", 2014 IEEE 17th International Conference.

[11] Minguel Grinberg, "Flask Web Development: Developing Web Applications with Python", 2018.