CSE 6940 (01) - GRAD RESEARCH METHODS IN CS
Prof. Said Ngobi

Name: Megha John Babu
Coyote Id: 008458288

Real Time Face-Recognition using OpenCV & Python

Introduction:
Face recognition is a widely used biometric technology that identifies or verifies a person's identity by analyzing their facial features. In this project, we implement a real-time face recognition system using OpenCV and Python. The system captures video from a webcam, detects faces, and matches them against a pre-trained dataset. It is a practical example of integrating computer vision techniques for real-world applications

Objective:
To develop a real-time face recognition system using OpenCV and Python that can detect and recognize faces from a live video feed.
Face detection is one of the most widely used computer vision applications and a fundamental problem in computer vision and pattern recognition.
OpenCV is a library which is used to carry out the image processing using programming languages (Python).
This project utilizes OpenCV using webcam

Software requirements:
Python 3.13.0
OpenCV
NumPy

System Components:
- Face Detection: Uses a Haar Cascade model to detect faces in each video frame
- Face Recognition: Recognizes detected faces by comparing them with a database of known faces using the LBPH (Local Binary Patterns Histograms) face recognizer

Methodology:
- Dataset Preparation: Collect and label images of individuals for training the recognizer
- Training the Model: Train the LBPHFaceRecognizer on the dataset, mapping each person to a unique ID
- Real-Time Video Processing:

Capture frames from the webcam

Convert each frame to grayscale for efficient processing
Detect faces in the frame and recognize them by comparing with trained IDs
Display recognized names and confidence scores on the screen

<u>The reason I chose this topic:</u>

- Real-time face recognition is a powerful and relevant technology with numerous practical applications
- This project allows for hands-on experience with computer vision techniques in Python and OpenCV, which are essential skills in both research and industry
- The topic also provides an opportunity to explore important concepts in machine learning and image processing, making it ideal for gaining insights into how artificial intelligence can be applied to solve real-world challenges
- Additionally, real-time recognition offers a unique technical challenge, as it requires both speed and accuracy to work effectively, adding depth to the learning experience

<u>For installing of python in the terminal:</u>

1. Python3 --version

It returns Python 3.13.0

2. python3 -m pip –version

It returns pip 24.2 from /opt/homebrew/lib/python3.13/site-packages/pip (python 3.13)

To install pip

3. Download pip

```
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
```

4. Add pip to path

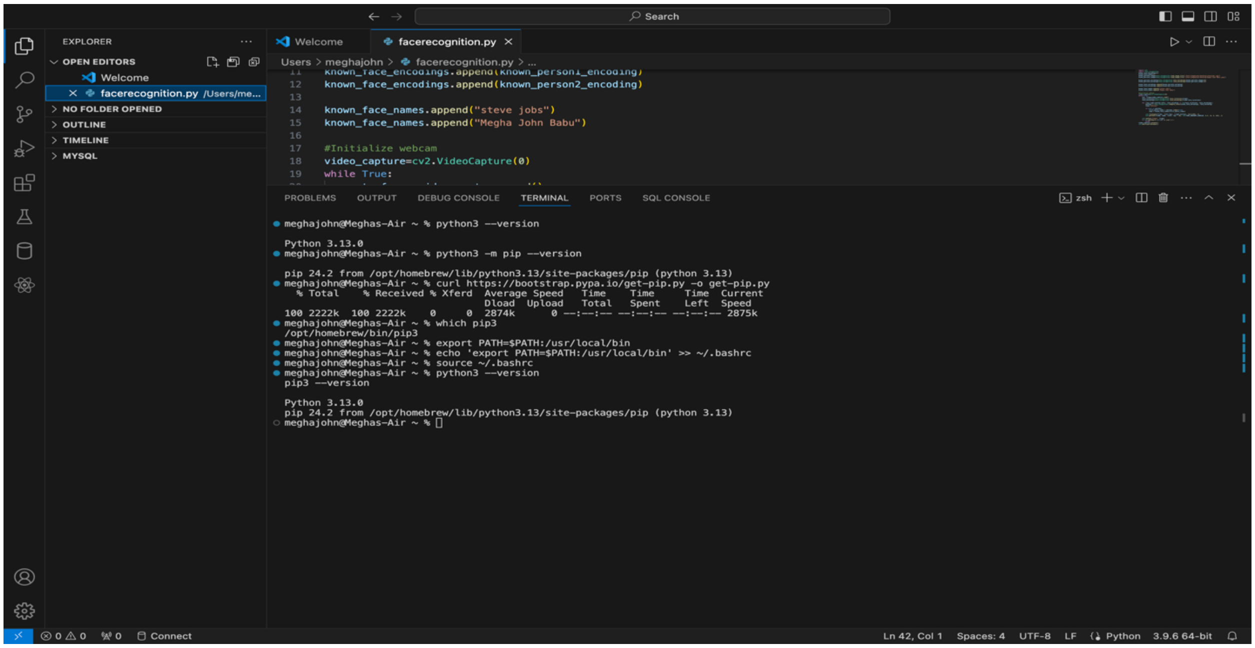
```
which pip3
```

5. Add its location to your PATH

```
export PATH=$PATH:/usr/local/bin
```

6. Add this line to your shell configuration file (~/.bashrc):

```
echo 'export PATH=$PATH:/usr/local/bin' >> ~/.bashrc
source ~/.bashrc
```



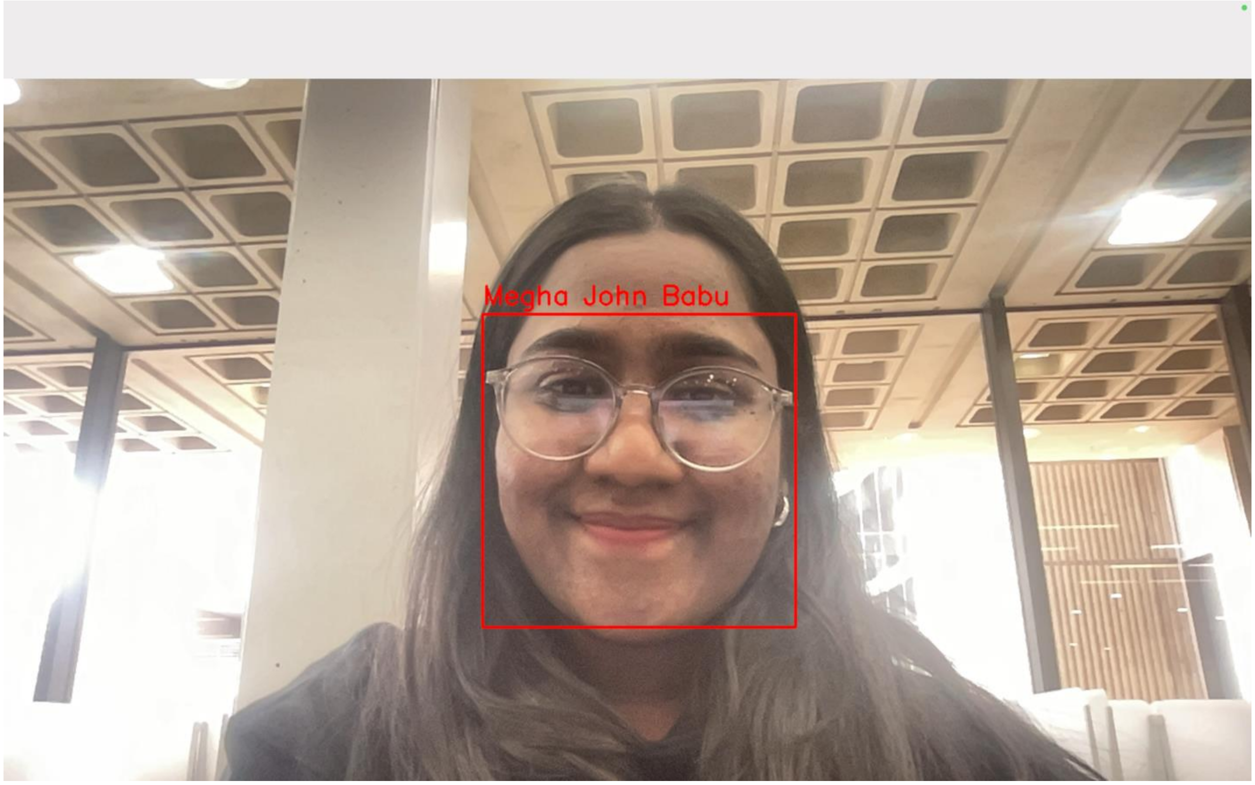
```
known_face_encodings.append(known_person1_encoding)
known_face_encodings.append(known_person2_encoding)

known_face_names.append("steve jobs")
known_face_names.append("Megha John Babu")

#Initialize webcam
video_capture=cv2.VideoCapture(0)
while True:
```

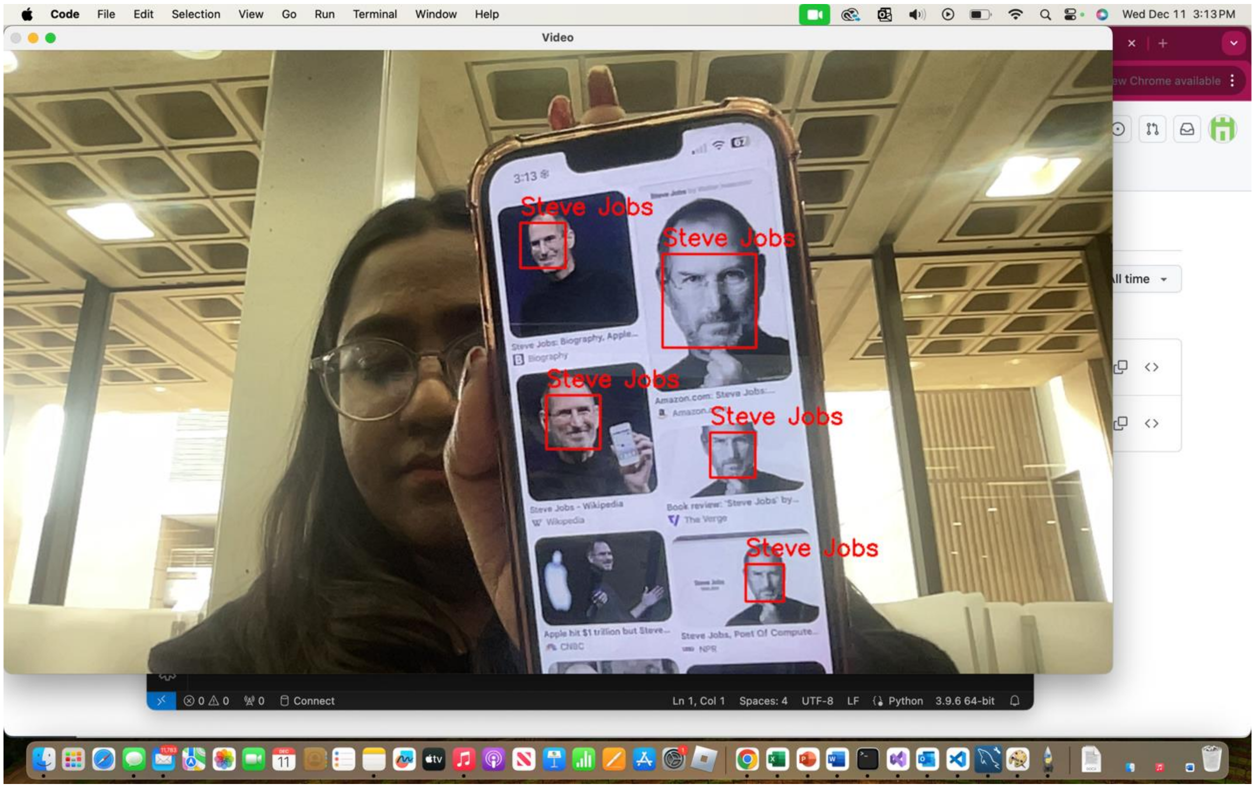
```
meghajohn@Meghas-Air ~ % python3 --version

Python 3.13.0
meghajohn@Meghas-Air ~ % python3 -m pip --version

pip 24.2 from /opt/homebrew/lib/python3.13/site-packages/pip (python 3.13)
meghajohn@Meghas-Air ~ % curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 2222k  100 2222k    0     0  2874k      0 --:--:-- --:--:-- --:--:-- 2875k
meghajohn@Meghas-Air ~ % which pip3
/opt/homebrew/bin/pip3
meghajohn@Meghas-Air ~ % export PATH=$PATH:/usr/local/bin
meghajohn@Meghas-Air ~ % echo 'export PATH=$PATH:/usr/local/bin' >> ~/.bashrc
meghajohn@Meghas-Air ~ % source ~/.bashrc
meghajohn@Meghas-Air ~ % python3 --version
pip3 --version

Python 3.13.0
pip 24.2 from /opt/homebrew/lib/python3.13/site-packages/pip (python 3.13)
meghajohn@Meghas-Air ~ % 
```

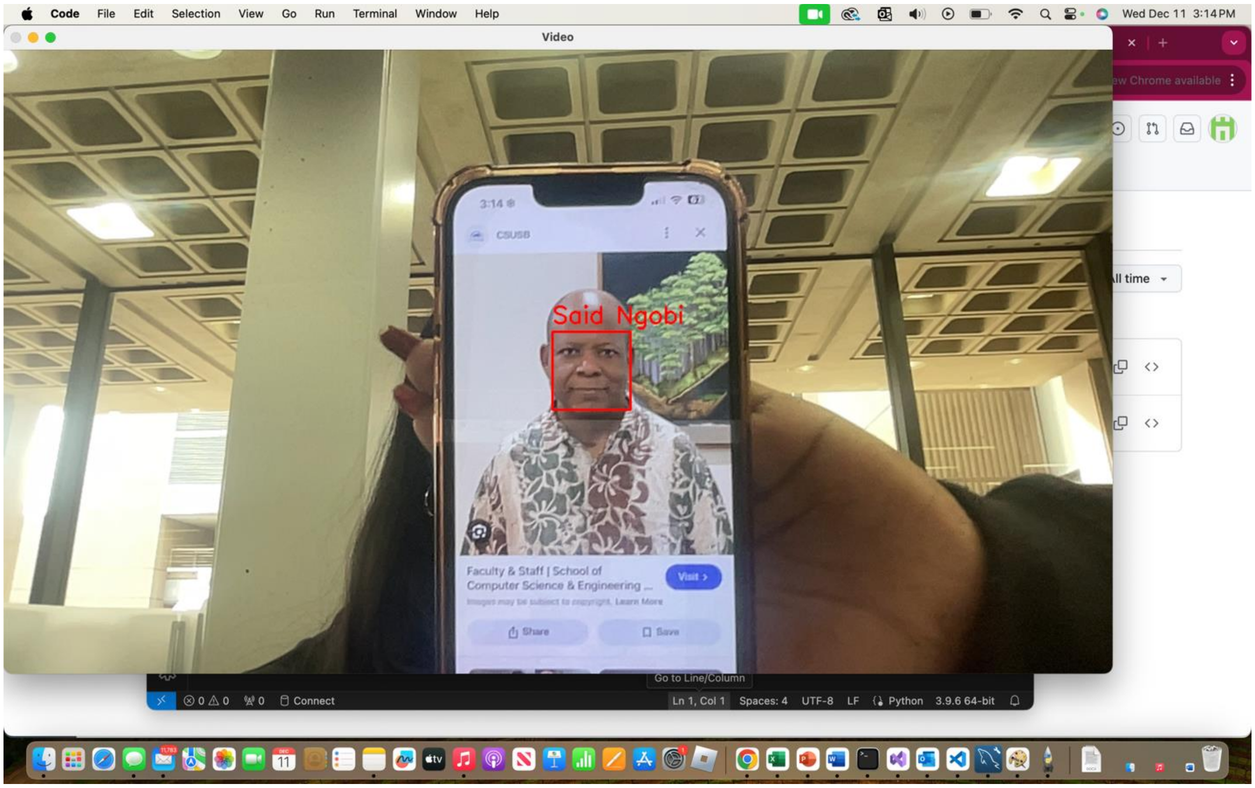
Implementation:

```
import cv2
import face_recognition
known_face_encodings=[]
known_face_names=[]
known_person1_image=face_recognition.load_image_file("/Users/meghajohn/Desktop/stevejobs.jpg")
known_person2_image=face_recognition.load_image_file("/Users/meghajohn/Desktop/Megha John Babu.jpeg")
known_person3_image=face_recognition.load_image_file("/Users/meghajohn/Desktop/said ngobi.jpg")

known_person1_encoding=face_recognition.face_encodings(known_person1_image)[0]
known_person2_encoding=face_recognition.face_encodings(known_person2_image)[0]
known_person3_encoding=face_recognition.face_encodings(known_person3_image)[0]

known_face_encodings.append(known_person1_encoding)
known_face_encodings.append(known_person2_encoding)
known_face_encodings.append(known_person3_encoding)

known_face_names.append("Steve Jobs")
known_face_names.append("Megha John Babu")
known_face_names.append("Said Ngobi")

video_capture=cv2.VideoCapture(0)
while True:
    ret, frame=video_capture.read()
    face_locations=face_recognition.face_locations(frame)
    face_encodings=face_recognition.face_encodings(frame,face_locations)

    for (top,right,bottom,left), face_encoding in zip(face_locations, face_encodings):
        matches = face_recognition.compare_faces(known_face_encodings, face_encoding)
        name="Unknown"

        if True in matches:
            first_match_index = matches.index(True)
            name = known_face_names[first_match_index]

        cv2.rectangle(frame, (left,top), (right,bottom), (0,0,255), 2)
        cv2.putText(frame, name, (left, top - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 0,
255), 2)

    cv2.imshow("Video", frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
video_capture.release()
cv2.destroyAllWindows()
```

<u>Screenshots of the Output:</u>




Code File Edit Selection View Go Run Terminal Window Help
Wed Dec 11 3:13 PM
Video
Steve Jobs
Steve Jobs
Steve Jobs
Steve Jobs
Steve Jobs
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit


Code File Edit Selection View Go Run Terminal Window Help
Wed Dec 11 3:14 PM
Video
Said Ngobi
Faculty & Staff | School of Computer Science & Engineering ...
Go to Line/Column
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.9.6 64-bit

Result:

- **Dataset**: Collected images for known individuals
- **Model**: LBPH Face Recognizer successfully identified individuals with reasonable accuracy
- **Performance**: The system displayed real-time recognition with minimal latency
- **Challenges**:
  - Recognition accuracy decreases with poor lighting or low-quality images
  - Limited dataset size affects performance

Conclusion:

The real-time face recognition system demonstrates how OpenCV and Python can be used to implement a practical biometric application. While effective for small-scale tasks, the system's accuracy can be improved with:

- Larger and more diverse datasets
- Advanced deep learning models like FaceNet or Dlib
- Preprocessing techniques for better feature extraction

Future Improvements:

- Integrate **Dlib** for more robust face detection and feature encoding
- Use a database like **SQLite** to manage recognized individuals
- Implement confidence thresholds for better handling of unknown faces
- Explore deep learning approaches for higher accuracy on larger datasets

References:

- OpenCV Documentation: https://opencv.org
- Python OpenCV Tutorials: https://docs.opencv.org
- Dlib: https://dlib.net