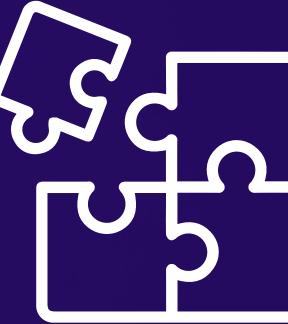


ADVANCED COMPUTER VISION



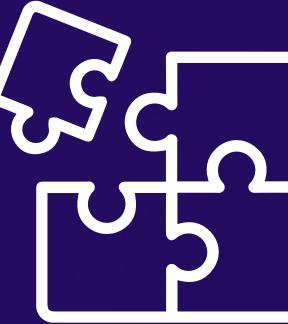


Guided By:
REGIN BOSE ME Phd

Submitted By: BATCH 51
S Mohan Kumar 210420104096
K Kishore 210420104081



OBJECTIVE



The scope of this project is to show gesture recognition with the goal of interpreting human gesture via mathematical algorithms. Gesture can originate from any bodily state but commonly originate from face and hand from which various calculations can be made by the computer



Introduction

Computer Vision focuses on enabling computers to interpret and understand visual information from the world around them, just as humans do.

Advancements in computer vision have paved the way for many exciting applications, including self-driving cars, facial recognition, and augmented reality. As technology continues to evolve, computer vision will play a vital role in shaping the future and improving our daily lives.





LITERATURE SURVEY

Hand gesture and Face recognition is a significant problem for human-computer interaction. One form of hand gesture recognition is static hand gestures. Feature extraction Module, Processing Module, and Classification Module. The feature extraction module uses human pose estimation with a top-down method to extract not only the keypoints but also body and hand bounding boxes. After being normalized and processed in the processing module, its output will be used as the input for the classification module in which we proposed an architecture

```
require 'capybara/rspec'  
require 'capybara/rails'  
  
Capybara.javascript_driver = :webkit  
Category.delete_all; Category.create  
Shoulda::Matchers.configure do |config|  
  config.integrate do |with|  
    with.test_framework :rspec  
    with.library :rails  
  end  
end
```



“

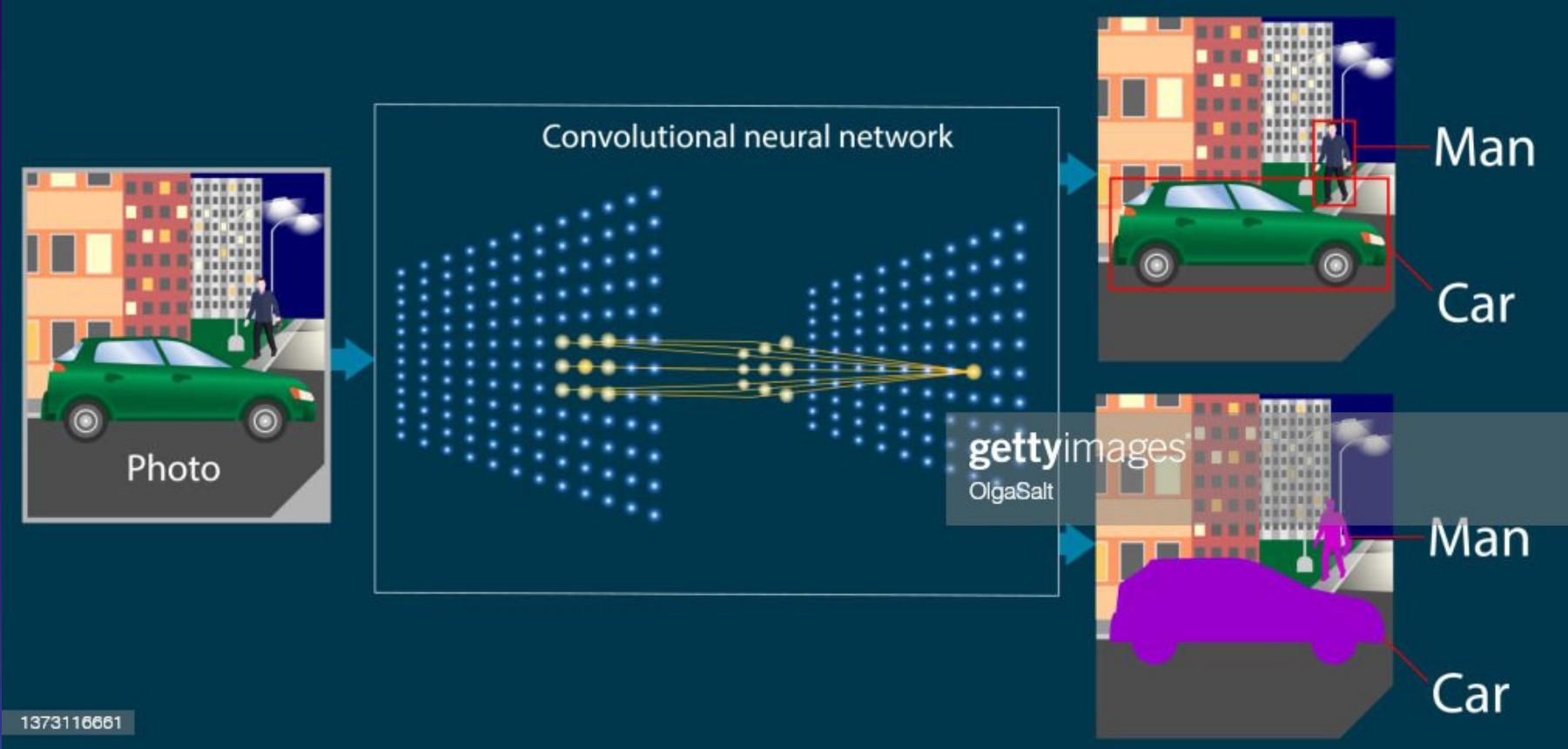
Existing System

- Drawbacks
- Solutions

EXISTING SYSTEM

There are many existing systems in computer vision that are used for various applications such as image recognition, object detection, image segmentation, and many others

Object detection and Instance segmentation



● Convolutional Neural Network(CNN)

This is one of the most widely used deep learning models for image classification and object detection.

● You Only Look Once(YOLO)

YOLO is a real-time object detection system that uses a single neural network to detect objects in an image.



Drawbacks

COMUTATIONAL COMPLEXITY

Computer vision algorithms, require a significant amount of computational resources, which can make real-time implementation challenging, especially on resource-constrained devices.

DATASET BIAS

The performance of computer vision algorithms is heavily dependent on the quality and diversity of the training data. If the training data is biased, the model will also be biased, leading to incorrect predictions.

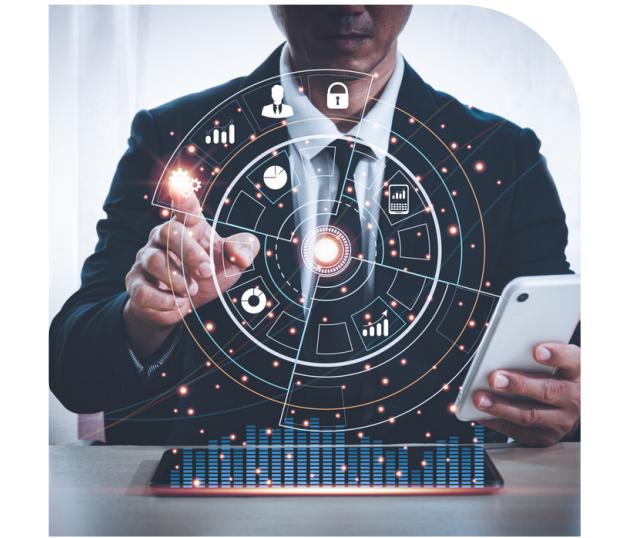


Solution



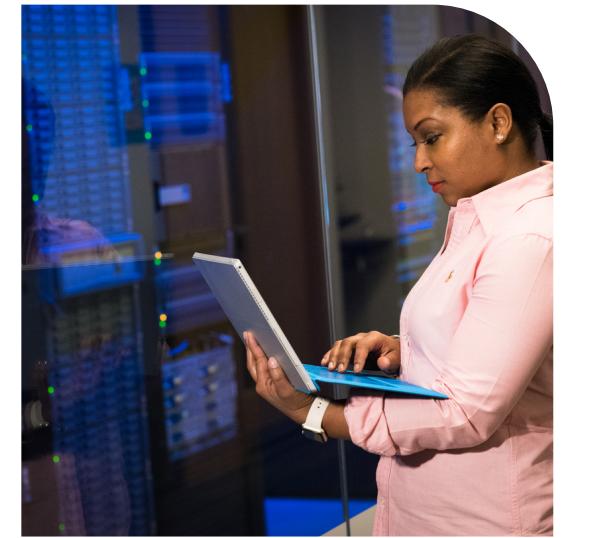
Image Processing

Edge detection



Feature Extraction

SIFT , SURF



Object detection

Object
Recognition

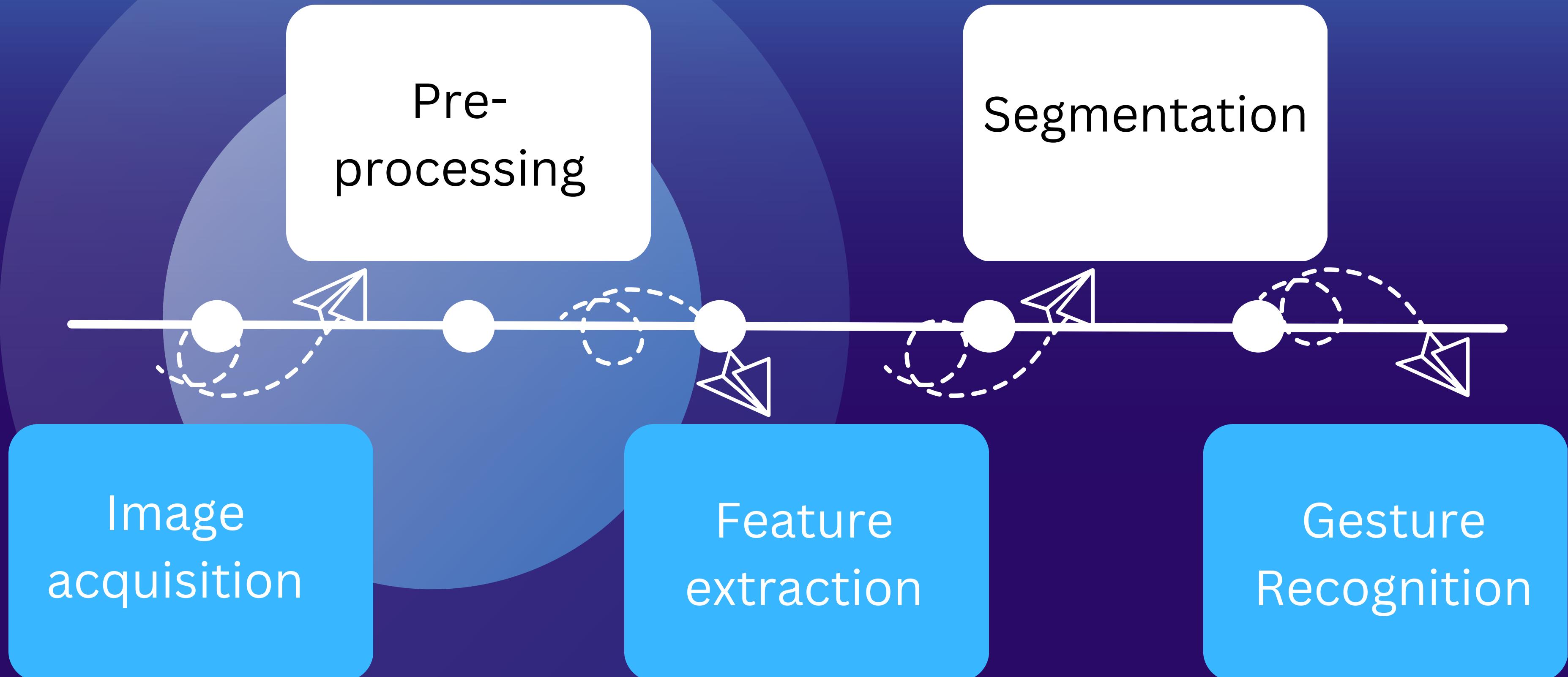


Image Segmentation

Constituent
Segments



Proposed System Block Diagram





Advantages

01

Increase and Decrease the System volume

Increase and Decrease the volume of the system automatically through hand gesture.

02

Face Recognition

Recognizes the face of the humans through their expressions.

03

Measures the FPS rate of the camera

Measures the FPS rate of the camera at a time and present in the screen.



Module Details

HANDTRACKINGMODULE

findHands()

findPosition()

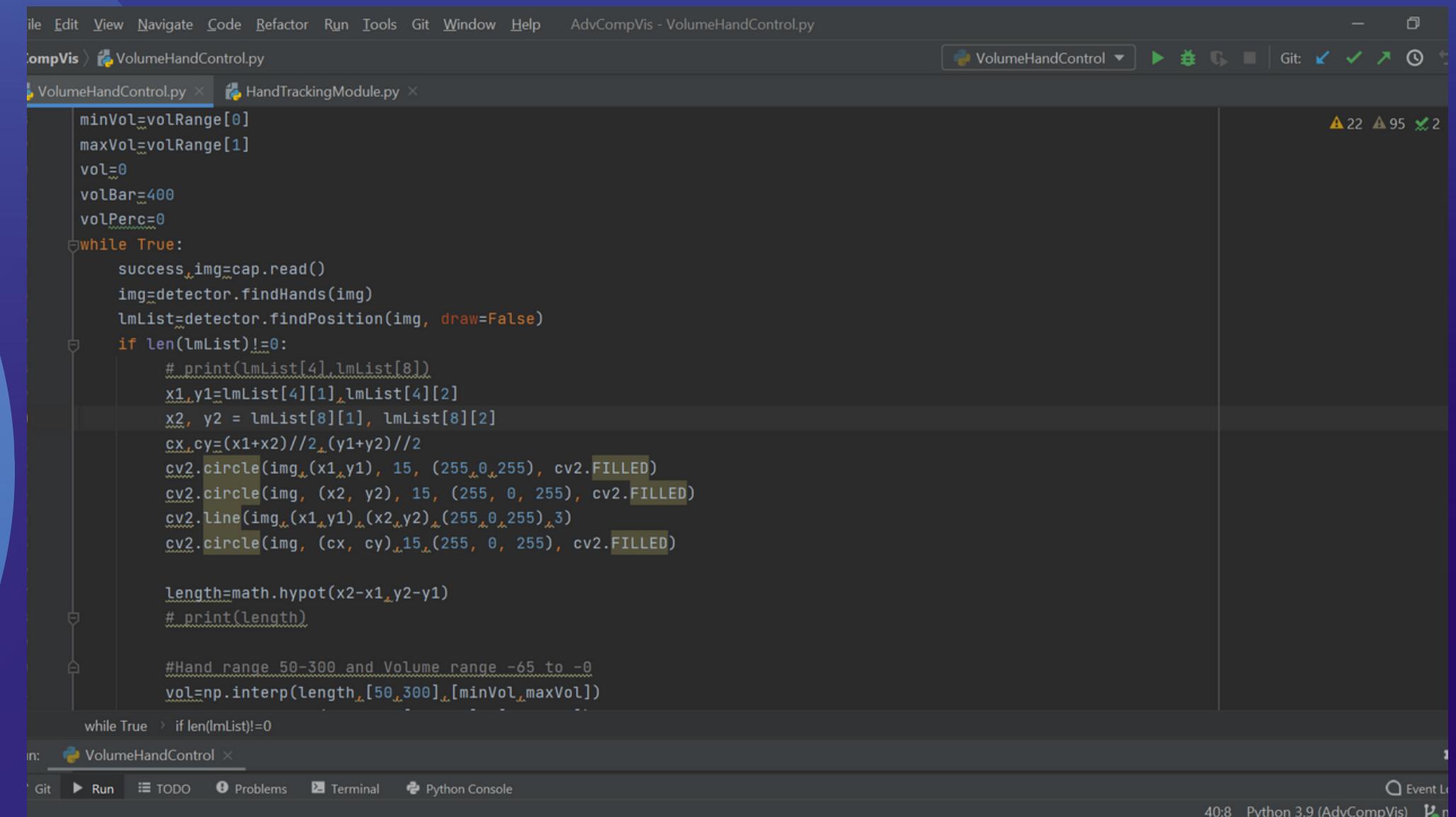
VOLUMEHANDCONTROMODULE

pycaw package

FACERECOGNITIONMODULE

findfaces()

fancyDraw()

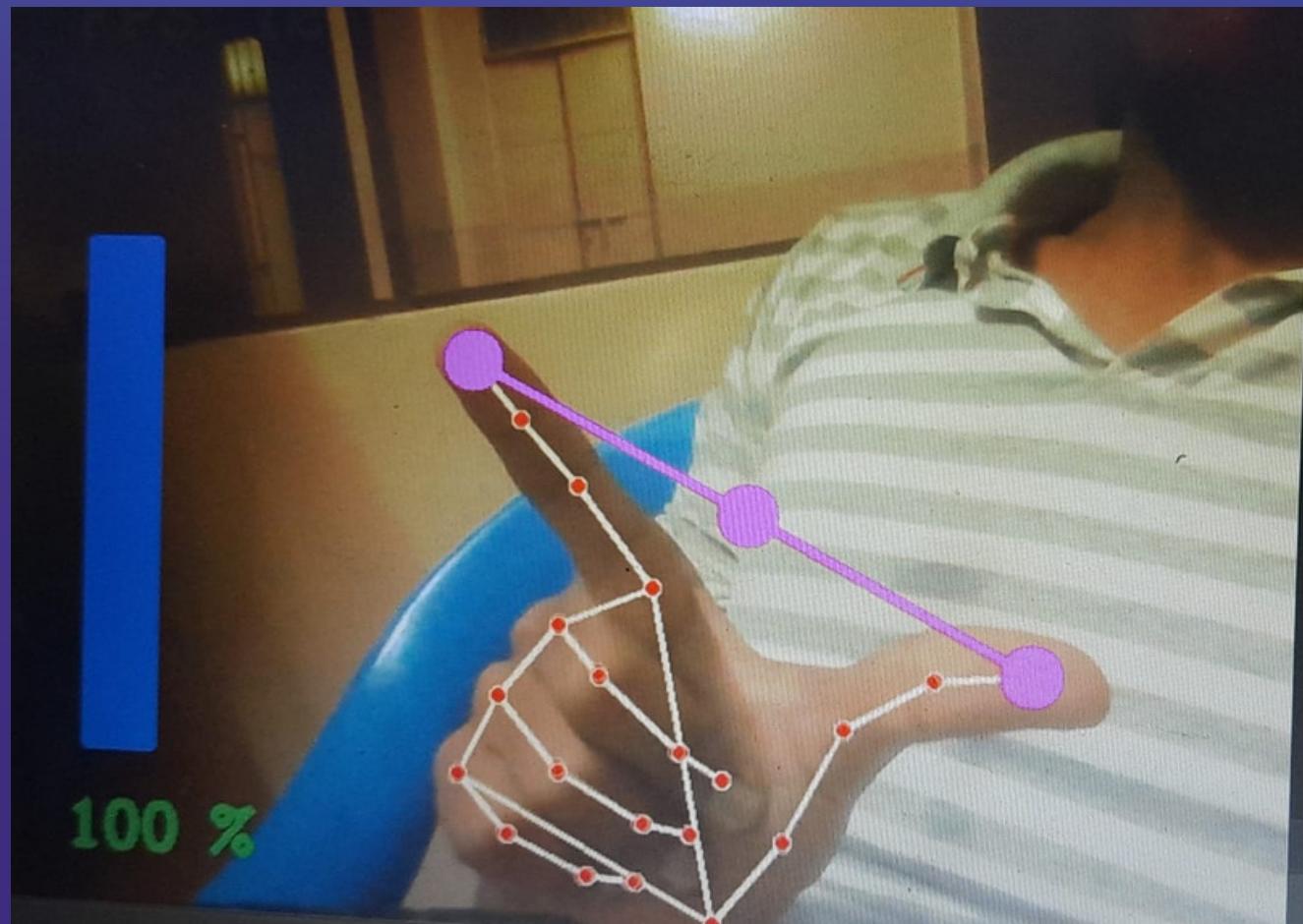


```
File Edit View Navigate Code Refactor Run Tools Git Window Help AdvCompVis - VolumeHandControl.py
CompVis > VolumeHandControl.py
VolumeHandControl.py x HandTrackingModule.py x
minVol=volRange[0]
maxVol=volRange[1]
vol=0
volBar=400
volPerc=0
while True:
    success,img=cap.read()
    img=detecto.findHands(img)
    lmList=detecto.findPosition(img, draw=False)
    if len(lmList)!=0:
        # print(lmList[4].lmList[8])
        x1,y1=lmList[4][1],lmList[4][2]
        x2, y2 = lmList[8][1], lmList[8][2]
        cx,cy=(x1+x2)//2,(y1+y2)//2
        cv2.circle(img,(x1,y1), 15, (255, 0, 255), cv2.FILLED)
        cv2.circle(img, (x2, y2), 15, (255, 0, 255), cv2.FILLED)
        cv2.line(img,(x1,y1),(x2,y2),(255,0,255),3)
        cv2.circle(img, (cx, cy),15,(255, 0, 255), cv2.FILLED)

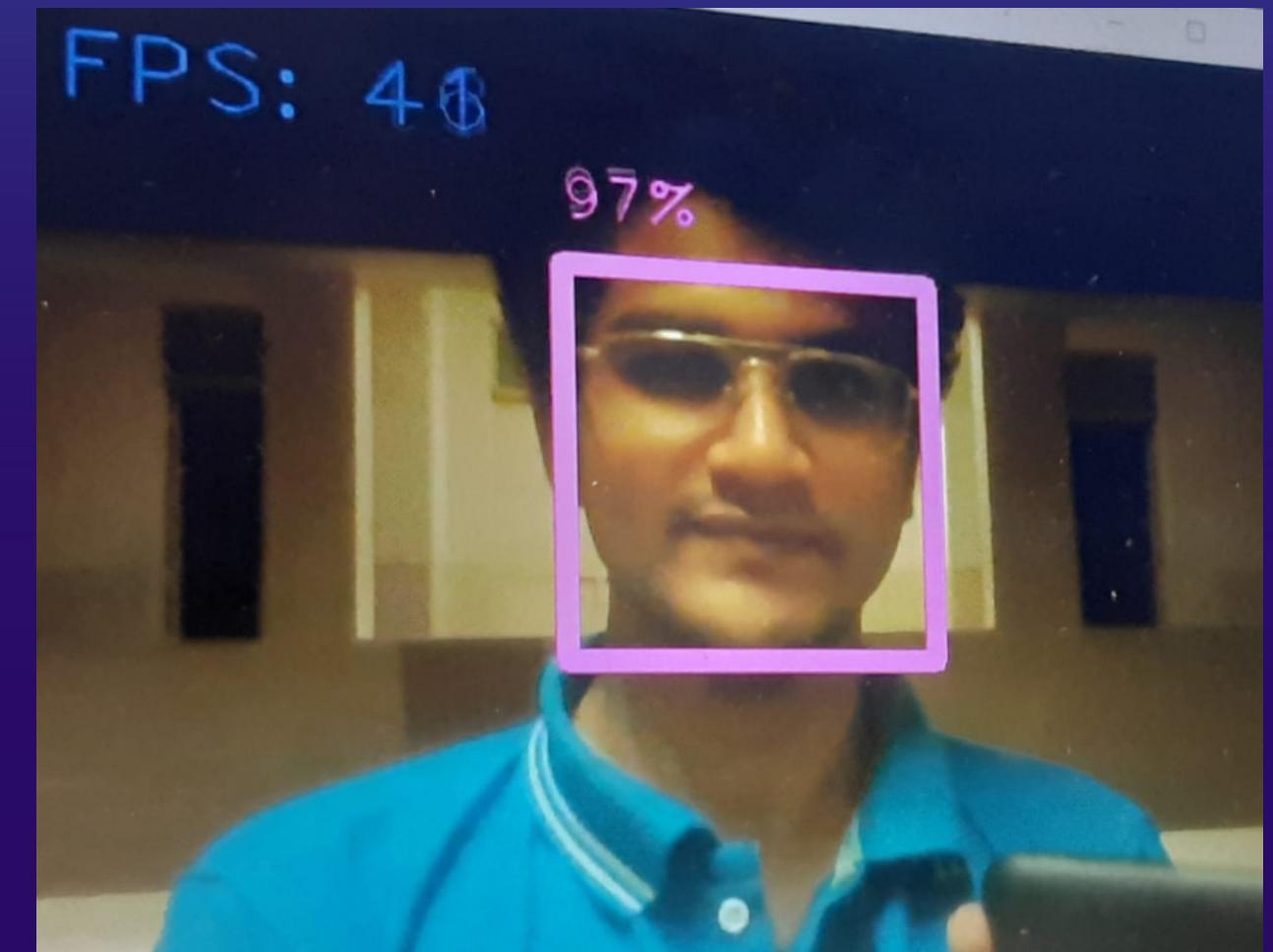
        length=math.hypot(x2-x1,y2-y1)
        # print(length)

        #Hand range 50-300 and Volume range -65 to 0
        vol=np.interp(length,[50,300],[minVol,maxVol])
    while True > if len(lmList)!=0
    in: VolumeHandControl x
    Git Run TODO Problems Terminal Python Console
    40:8 Python 3.9 (AdvCompVis) Event Log
```

Implementation



Volume Increase and
Decrease through Hand
Gesture



Recognizes Human Faces

REFERENCES

**FreeCodeCamp
YouTube**



**Science Hub
Science Direct**

©Team Delta
BATCH 51

Signs off

S MOHAN KUMAR

21420104096

K KISHORE

210420104081



Thank You.



9487047170



7994933210