Mini Project Third year on

Heartbeat Detection using Webcam

Submitted in partial fulfillment for the
Degree of Bachelor of Technology in
Data Science

Submitted by
Shreya Patel-2015005
Chinmai Rane-2015008
Suhasini Sharma-2015009

Under the guidance of
Prof. Rajesh Kolte

Usha Mittal Institute of Technology
S.N.D.T. Women's University
Juhu Tara Road, Santacruz(West),
Mumbai-400049

2022-23

CERTIFICATE

This is to certify that Shreya Patel, Chinmai Rane, Suhasini Sharma have completed the Mini Project Phase-II on the topic “ Heartbeat Detection using Webcam” in partial fulfillment for the Degree of Bachelor of Technology in Data Science under the guidance of Prof. Rajesh Kolte during the academic year 2022-2023 as prescribed by S.N.D.T. Women’s University, Mumbai.

Guide	Head Of Department
Prof.Rajesh Kolte	Prof.Rajesh Kolte

Principal

Dr.Shikha Nema

Examiner 1	Examiner 2

Acknowledgement

We would like to take up this opportunity to express gratitude to everyone who supported us throughout the course of this project. We are thankful for their aspiring guidance, in valuably constructive criticism and friendly advice during the project work. We are highly indebted to our project guide Prof. Rajesh Kolte for his guidance and constant supervision as well as for providing necessary information regarding the project also for his support in working towards the project. We would like to express our gratitude to our overall mini project coordinator Ms Sonali Bodekar who has guided us through the entire process of the project.We also express our sincere gratitude to our friends and family who provided us with the facilities being required and conductive conditions for our project and being a part of the literature survey.

Abstract

Heart rate measurement is valuable information in the description of an individual's condition. It's a contact-less method of measuring heart rate. The goal of heartbeat detection using webcam majorly focuses on pulse detection and its application in most commonly used electrical devices webcam pulse detector: It focuses on pre-frontal lobe (i.e. forehead region) to measure the optical intensity which is responsible for pulse measurement .Here we have used face detection, filtering and peak detection in the FFT domain. A stable heartbeat can be isolated in about 15 seconds when there is good lighting and minimum noise. After the estimation of the user's heart rate, real-time phase variation associated with this frequency is also computed and then, using a physiological signal called a photoplethysmograph (PPG), we use pre-established formulae and methods that we use to compute the patient's heart rate. It uses OpenCV for face detection and python for programming/coding.Support for detection on more than one individual at a time in a single camera's image is also possible. Thus enabling a smart health care system this project can help save the lives of patients in nick time.

Contents

List of Figures

Chapter 1

Introduction

As a result of sitting down jobs, a lack of exercise, and unhealthy eating habits, there are more and more cardiovascular diseases in the population today, which increases the number of deaths each year. However, many pieces of equipment that are used to measure heartbeat are expensive and not readily available in many locations or even available 24/7, so this project will use a webcam to take live snapshots of your face and by calculating the variation in colors that your forehead unites, it will be possible to measure heartbeat. A simple camera or webcam will take a live video of you while focusing on your forehead, changing the image or dividing it into RGB, but only the green image is used to calculate your heart rate. Our major objective is to develop a system that uses a consumer-grade camera to assess your heart rate. We will display heart rate after computing them in real-time using the camera. An online pulse monitor is a self-monitoring device that enables real-time heart rate measurement and display. It is primarily used to collect heart rate information while engaging in various forms of physical activity. Consumer-grade heart rate monitors are wire-free and made for regular use. The patient's face is first detected, and then, using a physiological signal called a photoplethysmograph , we use pre-established formulae and methods that we took from the study material that is provided below to compute the patient's heart rate.

1.1 Problem Statement

Presently, a lot of people pass away or suffer from significant illnesses including abrupt fainting after exercise, high blood pressure, cardiac issues, heart attacks, and other conditions that affect our pulse rate. As a result, various issues have been identified, including:

1. People may not be concerned about their health and are unaware of it.
2. Those who are busy and might be reluctant to visit the hospital for health screenings.
3. More people, young or old, are developing significant illnesses like high blood pressure, heart conditions, and other dangerous health issues. So, to address this issue, the system can inform the user of their health. By monitoring, the user will be aware of their health status. So, it will notify the user if their pulse rate measurement is normal or abnormal when an output is displayed on a laptop. The user can then take appropriate action.

Chapter 2

Literature Survey

2.1 Technical Papers

1. **Rallapalli, Archana. (2013). Measuring the pulse rate by using webcam**
   This paper developed a simple method for determining the physiological parameters of the face of different subjects using a Logitech webcam and PPG systems.

2. **Jain, Monika & Deb, Sujay Subramanyam, A.V. (2016). Face video based touchless blood pressure and heart rate estimation**
   Hypertension is a leading cause of premature deaths due to cardio-vascular diseases, and this paper proposes a novel touchless approach to predict BP and HR using face video based PPG.

3. **Bhanu Pratap, Surabhi Gautam , Naimishi Gupta, Saloni Gupta CSE Department, Inderprastha Engineering College Heartbeat Rate Monitoring System by Pulse Technique**
   Measurement of heart rate using webcam or phone's camera is a non-contact method that can help people care about their health and reduce the cost of a machine.

4. **Rahman, Hamidur & Ahmed, Mobyen Begum, Shahina Funk, Peter. (2016). Real Time Heart Rate Monitoring from Facial RGB Color Video Using Webcam**
   This paper presents a real time HR monitoring method using a webcam of a laptop computer, using signal processing methods such as Fast Fourier Transform (FFT), Independent Component Analysis (ICA) and Principal Component Analysis (PCA).

5. **Maximillian F. Xavier, Otavio A. Dias & Eduardo Peixoto (2016) Software for real time heart rate detection using a standard webcam**
   This work proposes a method to estimate a subject's heart rate using a standard camera based on subtle color variations caused by blood flow. Results are within 5 bpm of the correct heart rate.

6. **Measuring Heart Rate from Video Isabel Bush Stanford Computer Science 353 Serra Mall, Stanford, CA 94305**
   This paper re-implements GrabCut to measure heart rate in color video of a person's face. Facial segmentation improved the robustness of the algorithm to bounding box noise.

Chapter 3

Existing System

1. **Electromagnetic-based monitoring systems**
   The basic principle of radar is to transmit a microwave (radio) signal towards a target. The strength of the back scattered signal is measured. There are two variants of radar sensing used for heart rate monitoring: continuous-wave (CW) and wide band pulsed radar (UWB).
   At present, electrocardiography (ECG) is a well-established method for HR measurement.

2. **Laser-based monitoring systems**
   The LDVi is a non contact laser-based monitoring method, named vibrocardiography, VCG, to measure heart beat as well as the respiration activity , the heart mechanics and the artery stiffness.
   However, at the moment qualitative results are reported for only laboratory tests with special experimental set-up in ideal measurement conditions and therefore can't be considered a solution to the problem of direct contact between the device and the patient's skin for HR measurement.

3. **Image-based monitoring systems**
   These are based on the fact that the deflection of a human vessel due to heartbeats can be visually observed from the consequent deflection on the surrounding skin. Cardiocam is one of the most recent and promising

technologies for measurement of physiological signals such as heart rate and breathing rate using a basic digital imaging device such as a webcam.
Still the robustness of the system and the efficiency of the procedure on large scale needs to be improved as well as further investigation needs to be carried out for investigating the influence of factors such as movement, skin thickness, skin pigmentation and blood pressure, etc.

4. **Other methods**
   These methods for non contact monitoring of the heart include using magnetic induction, capacitive method and pressure oscillations.
   However, data reported are very sensitive to body movements which makes the proposed method needing further improvement before the use on a large scale.

Chapter 4

Proposed System

The study in this field is very broad, and there are many prospects to learn new methodologies for refining accuracy. The methodology we propositioned is based on the same ideologies as these research journals, with alterations to increase accuracy and achieve positive results. In order to decide which study article would provide the best outcomes, we viewed the advantages and disadvantages of each. The objective is to dodge methods that have led to decrease in accuracy.

4.1 Proposed Framework

Our proposed framework is shown in Fig. 4.1. It includes the following steps: frame capture, face detection and region-of-interest (ROI) definition, trace extraction, frequency domain filtering, peak detection and recurring point of interest analysis, heart rate display, and software implementation.

Program Workflow:

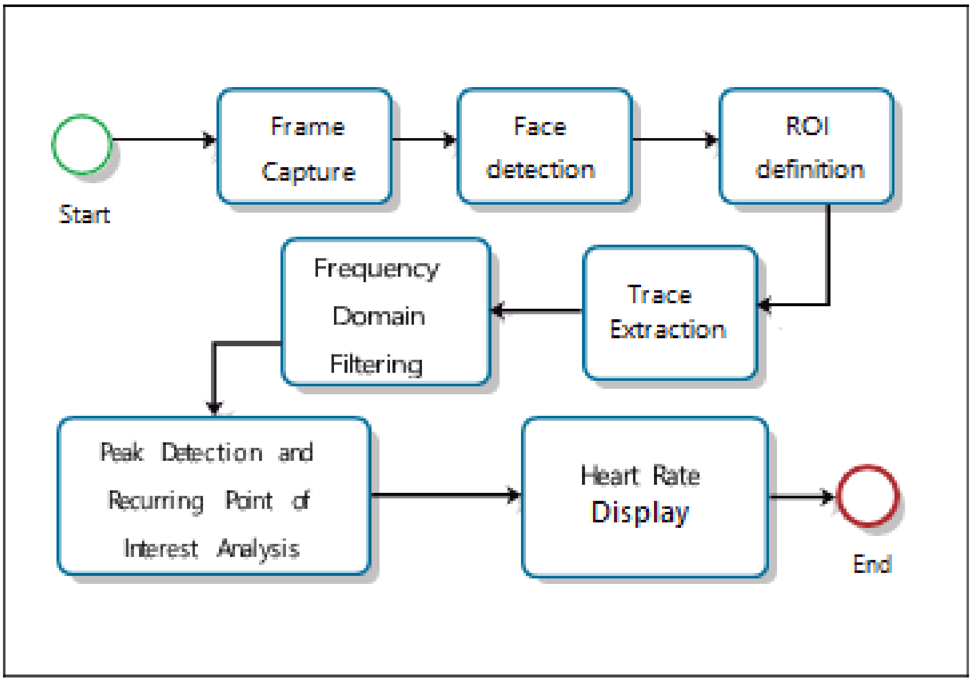


Figure 4.1: Program Workflow

- *Frame capture:* The first step involves the achievement of a frame through a webcam connected to the computer. The apprehended frame is in the RGB colour setup. It is essential that the webcam is capable of seizing the raw data, without any compression.

- *Face Detection and ROI Definition:* The second step contains facial recognition over the attained frame. By default, only one face is to be detected in this project. The third step entails describing a bounding box above the formerly detected face. The ROI is defined as h x w, where h is the height of the bounding box of the detected face and w is 55% of the width of this bounding box. The resulting ROI is to be studied in the next stage. The ROI is kept the same for the residual frames.

- *Trace Extraction:* In the fourth step, for each frame n, the ROI consists of a colour image h x w with three frequencies, Red, Green and Blue. This image is standardized, eliminating the mean value for every frequency, creating the variance between samples unitary as follows:

  $$\hat{C}_i(n) = \frac{C_i(n) - \mu_i}{\sigma_i} \tag{4.1}$$

  Here i is for R, G and B, μ_i and σ_i are the mean and standard deviation for each frequency, respectively.

- *Frequency Domain Filtering:* In the fifth step, the signal of interest is the trace, T (n). Algorithms are not practical for each new trace sample, but relatively once every two seconds. The algorithm takes the trace values for the last 10 seconds and stores them in a vector. It is then changed to the frequency domain with a Fast Fourier Transform - FFT of 2048 points.

- *Peak Detection and Recurring Point of Interest Analysis:* This is the sixth step, which executes the heart rate estimation by studying the prevailing peaks in the found signal in the prior step.

- *Heart Rate Display:* The last step comprises putting on a poignant average filter of size 3 i.e., the outcome exhibited on the screen is an average value between the last 3 estimates.

Chapter 5

Requirements

5.1 Software Requirements

VS Code-Visual Studio Code is being used as it is a free source code editor that fully supports Python and useful features such as real-time collaboration.

Python Libraries-Python libraries like OpenCV,Flask,numpy were used until now for video stream access, face detection and streaming on webpage.

5.2 Hardware Requirements

Laptop-Any kind to run the code and build the model using VS Code.

HD Webcam-To access the live video stream for further processing.

Chapter 6

Implementation

1. Accessed the video stream of local webcam using OpenCV-We can record video from the camera using OpenCV. It enables you to build a webcam video capture object that you can use to record videos and then manipulate them as you see fit. cv2.VideoCapture() is used to get a video capture object for the camera.read() method is used to read the frames using the above created object.imencode() function is used to (encode) format frames from video into streaming data and stores it in-memory cache.

2. Streamed the video stream on web page using Flask web framework-The Flask library was imported to implement the web application, and @app.route is a Python decorator that tells the @app to execute the index() function when a user visits the domain. run() is called and the web-application is hosted locally on [localhost:5000]. Debug=True ensures that changes are made while the server is still running.

3. Implemented Face recognition using Haar-Cascade of OpenCV-Haar Cascade is a feature-based object detection algorithm that uses pre-trained classifiers to detect objects from images. It breaks down an image into smaller subregions and scans each region for patterns that match specific facial features. If enough potential features are detected, the algorithm identifies the image as a face.

4. Converted the frame to grayscale and forehead ROI isolation-Coverting to grayscale helps in simplifying algorithms and as well eliminates the complexities because grayscale compressors an image to its barest minimum pixel.Passed the frame to the cv2.cvtColor() method with color gray as a parameter to convert it into gray-scale.ROI isolation was done so that a portion of an image that we want to filter or operate on in some way is only chosen and to avoid the processing of irrelevant image points and accelerate the processing.

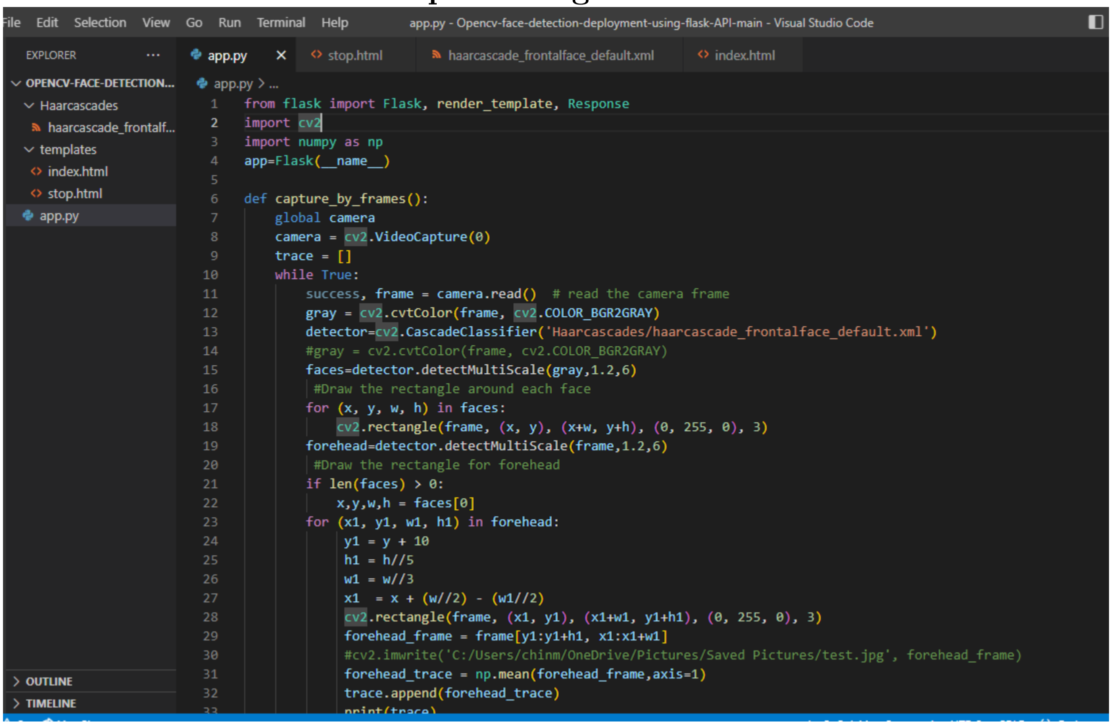
5. Trace extraction-Trace extraction is done as it will be helpful for next steps.We got all the RGB traces to detect the changes in blood flow as when blood flows there is small change in skin color which can be analysed by using RGB traces found from forehead region.

6. Frequency Domain Filtering-The frequency domain filtering is achieved using the Fast Fourier Transform (FFT) to convert the intensity variations over time into a frequency domain representation. The FFT decomposes the signal into its constituent frequencies and their amplitudes.To filter out noise and isolate the heart rate frequency components, a band-pass Butterworth filter is applied to the FFT coefficients.After applying the band-pass filter, the filtered signal is transformed back to the time domain using the inverse FFT.

7. Peak Detection and Recurring Point of Interest Analysis-After filtering the signal, the next step is to detect peaks in the signal. Peaks refer to the points of the signal where the amplitude is higher than its neighboring points.Once the peaks are detected, the next step is to calculate the BPM (beats per minute) using the time interval between consecutive peaks.

8. Heart Rate Display-In the heart rate display part, the bpm variable, which stores the calculated heart rate, is displayed on the video frame. This is achieved by drawing a text on the video frame using the OpenCV putText function.

Pre-processing of Model:



```python
from flask import Flask, render_template, Response
import cv2
import numpy as np
app=Flask(__name__)

def capture_by_frames():
    global camera
    camera = cv2.VideoCapture(0)
    trace = []
    while True:
        success, frame = camera.read()  # read the camera frame
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        detector=cv2.CascadeClassifier('Haarcascades/haarcascade_frontalface_default.xml')
        #gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces=detector.detectMultiScale(gray,1.2,6)
        #Draw the rectangle around each face
        for (x, y, w, h) in faces:
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 3)
        forehead=detector.detectMultiScale(frame,1.2,6)
        #Draw the rectangle for forehead
        if len(faces) > 0:
            x,y,w,h = faces[0]
        for (x1, y1, w1, h1) in forehead:
            y1 = y + 10
            h1 = h//5
            w1 = w//3
            x1  = x + (w//2) - (w1//2)
            cv2.rectangle(frame, (x1, y1), (x1+w1, y1+h1), (0, 255, 0), 3)
            forehead_frame = frame[y1:y1+h1, x1:x1+w1]
            #cv2.imwrite('C:/Users/chinm/OneDrive/Pictures/Saved Pictures/test.jpg', forehead_frame)
            forehead_trace = np.mean(forehead_frame,axis=1)
            trace.append(forehead_trace)
            print(trace)
```

Figure 6.1: Pre-processing code

Pre-processing of Model:

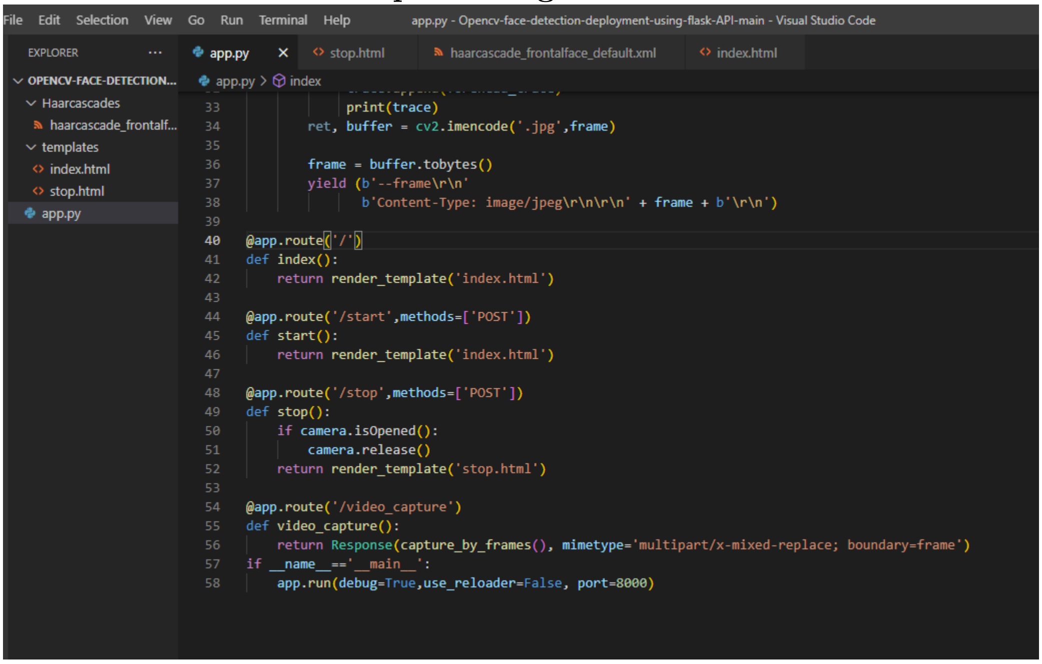


Figure 6.2: Pre-processing code

Pre-processing of Model:Output 1

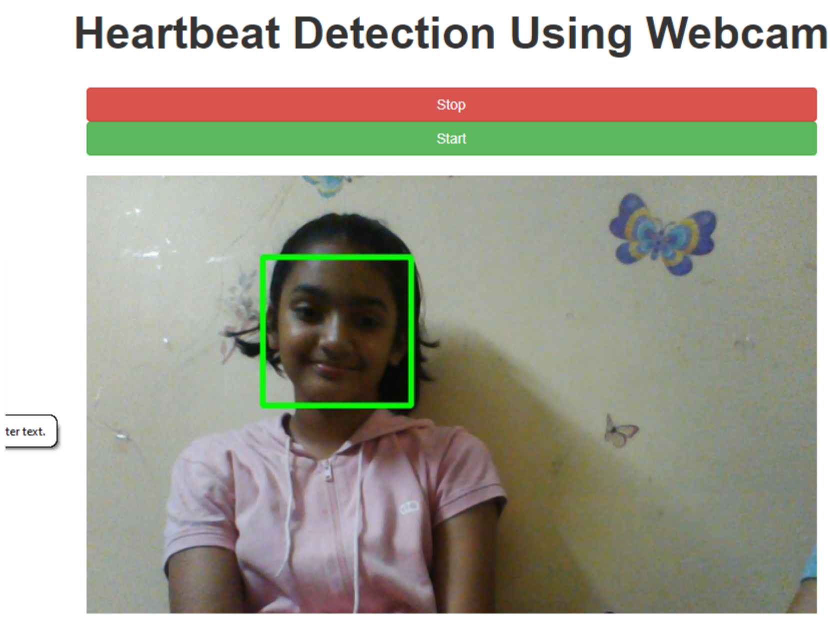


Figure 6.3: Video stream and Facial Recognition

Pre-processing of Model:Output 2

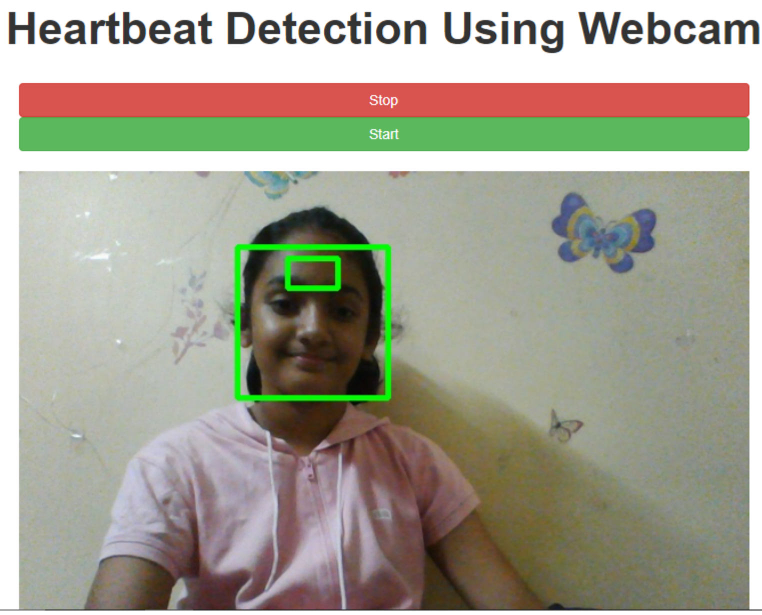


Figure 6.4: Bounding box for Forehead

Pre-processing of Model:Output 3

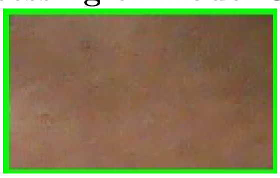


Figure 6.5: Forehead ROI isolation

Pre-processing of Model:Output 4

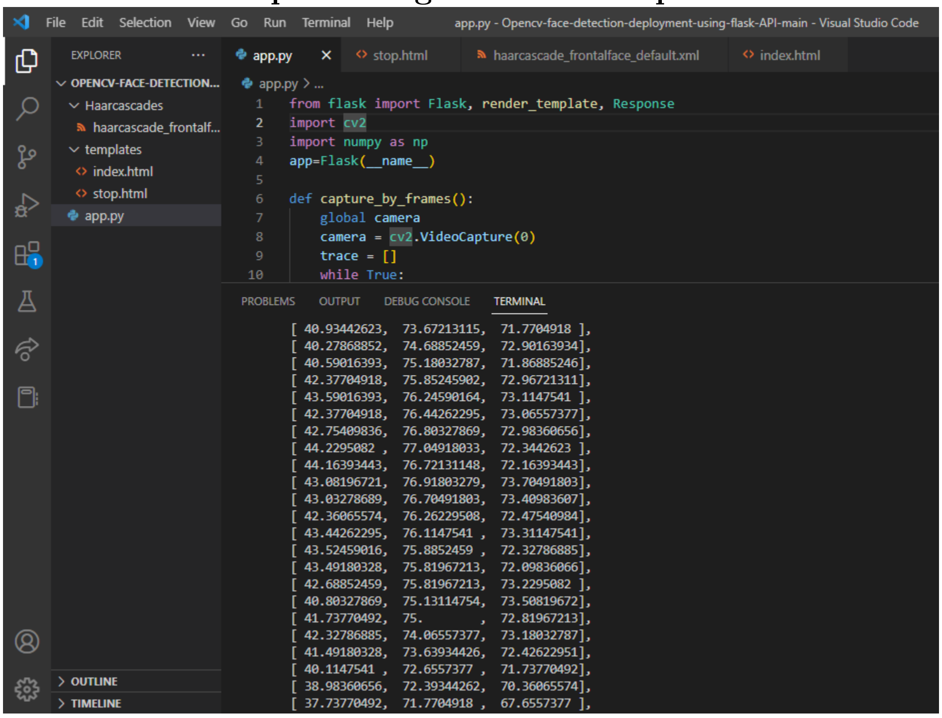


Figure 6.6: Trace extraction output

Face Detection

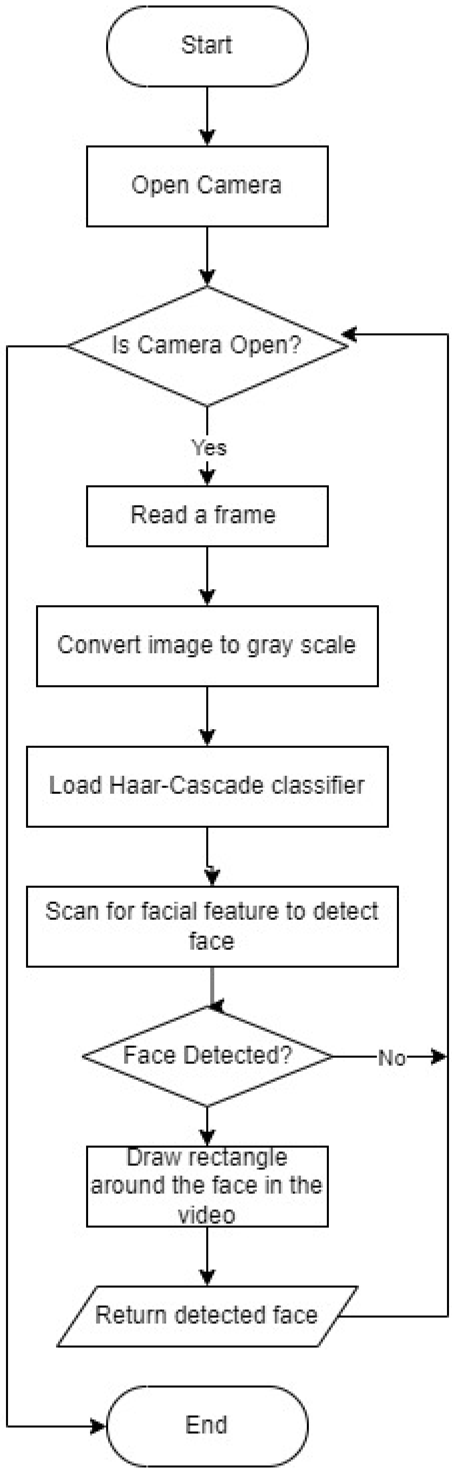


Figure 6.7: Working of Haar-Cascade classifier

Face Detection

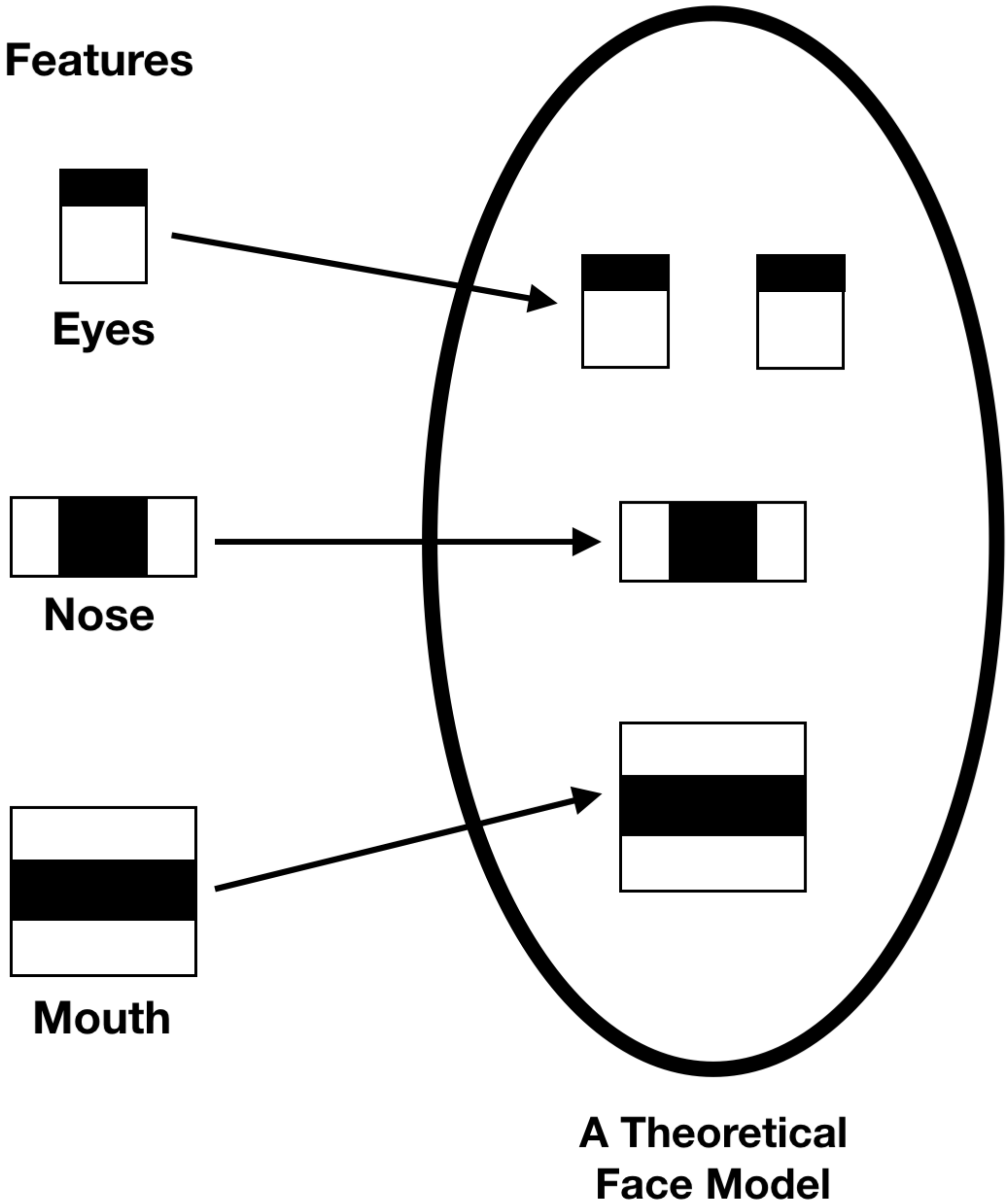


Figure 6.8: Face recognition using Haar-Cascade

Pre-processing of Model:

```
std = np.std(forehead_trace)
normalized_trace = (forehead_trace - mean) / std
if not np.log2(len(normalized_trace)).is_integer():
    padded_length = int(2**(np.ceil(np.log2(len(normalized_trace)))))
    normalized_trace = np.pad(normalized_trace, (0, padded_length-len(normalized_trace)), mode='const
    trace.append(normalized_trace)
    #print(normalized_trace)
#print(forehead_trace)
#print(trace)

fft_signal = fft(normalized_trace)
freqs = fftfreq(len(normalized_trace))
#print(freqs)
fft_signal_filtered = fft_signal.copy()
# Define the filter parameters
low_cutoff = 0.75
high_cutoff = 5
filter_order = 1

# Define the sampling rate
fs = 25

# Create a filter
b, a = butter(filter_order, [low_cutoff / (fs / 2), high_cutoff / (fs / 2)], btype='band')

# Filter the signal
filtered_forehead_trace = lfilter(b, a, forehead_trace)
```

Figure 6.9: Pre-processing code

Pre-processing of Model:

```
# Calculate the BPM
if len(peaks) > 0:
    bpm = int(60 / np.mean(np.diff(peaks)) * fs * len(forehead_trace) / (high_cutoff - low_cutoff))
    print("BPM:", bpm)
else:
    print("No peaks detected.")
```

Figure 6.10: Pre-processing code

Pre-processing of Model:

```
for (x, y, w, h) in faces:
    cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 3)
    cv2.putText(frame, f'BPM: {bpm}', (x, y-10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 0, 255), 2)
forehead=detector.detectMultiScale(frame,1.2,6)
```

Figure 6.11: Pre-processing code

Final Output of Model:Output 3

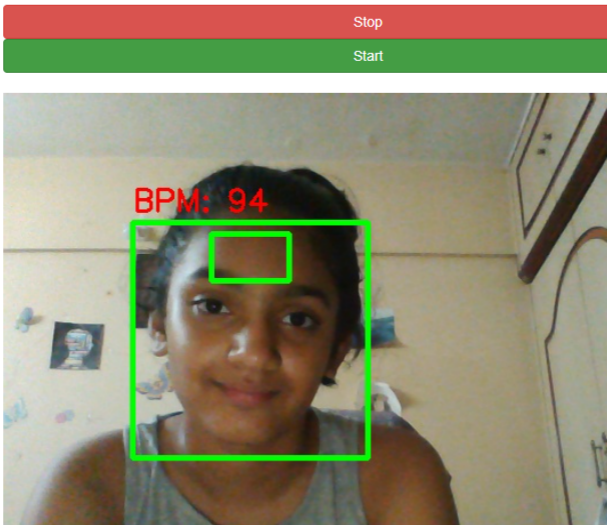


Figure 6.12: Heartrate Display

Chapter 7

Applications

1. It aids telehealth, also referred to as telemedicine or e-medicine, which is the remote delivery of healthcare services over the telecommunications infrastructure.

2. This non-contact technology is promising for medical care and other indoor applications due to widespread availability of cameras, especially webcams.

3. They can be used to estimate and analyze a person's physical health, detect possible diseases, and monitor recovery.

4. Non-contact detection and monitoring of human heart rate is a reliable and stable methodology that can be extended to a wide range of applications in the fields of emergency and military healthcare.

Chapter 8

Future Scope

1. Attain a highest level of accuracy, which has yet to be obtained for this model.
2. To increase the efficiency, the experiment needs to be done by more test subjects and more verifying systems.
3. Many other important physiological parameters such as RR, HRV and arterial blood oxygen saturation can potentially be estimated using the proposed technique.
4. Creating a real-time, multi-parameter physiological measurement platform with higher resolution of video based on this technology in driving situations will be the subject of future work.

Chapter 9

References

[1] Rallapalli, Archana. (2013). Measuring the pulse rate by using webcam.

[2] Jain, Monika & Deb, Sujay & Subramanyam, A.V. (2016). Face video based touchless blood pressure and heart rate estimation.

[3] Bhanu Pratap, Surabhi Gautam , Naimishi Gupta, Saloni Gupta CSE Department, Inderprastha Engineering College Heartbeat Rate Monitoring System by Pulse Technique.

[4] Rahman, Hamidur & Ahmed, Mobyen & Begum, Shahina & Funk, Peter. (2016). Real Time Heart Rate Monitoring from Facial RGB Color Video Using Webcam.

[5] Maximillian F. Xavier, Otavio A. Dias & Eduardo Peixoto (2016) Software for real time heart rate detection using a standard webcam.

[6] Isabel Bush (2016) Measuring Heart Rate from Video.