School of Science and Technology
University of Siegen

Remote Photoplethysmography using Commercially Available Webcams

Studienarbeit

Mechatronics

Kaushik Goud Chandapet

Nov 9th, 2021

Examiners
Prof. Dr. Kristof Van Laerhoven — School of Science and Technology
M. Sc. Florian Wolling — School of Science and Technology

Adviser
M. Sc. Florian Wolling

This Studienarbeit is handed in according to the requirements of the University of Siegen for the study program Master of Science (M. Sc.) Mechatronics.

Process period
May 6th, 2021 to Nov 9th, 2021

Examiners
Prof. Dr. Kristof Van Laerhoven
M. Sc. Florian Wolling

Adviser
M. Sc. Florian Wolling

Declaration of Originality

I hereby declare that this Studienarbeit is my original work and only the indicated sources and resources were used. All references, ideas, and direct quotes taken from other sources and used in my report have been fully and properly cited.

Furthermore, it does not contain any material previously published or submitted, neither in whole nor in part, for credit of any other degree at any institution.

....	....
Place, Date	Signature

Contents

1 Introduction

Heart rate (HR) is influenced by the autonomic nervous system which consists of two primary branches, the sympathetic and the parasympathetic nervous system [8]. The sympathetic nervous system is active when we are in an exciting situation (threat, fear, stress, and exercise) and increases heart rate whereas the parasympathetic nervous system is active when we are in a relaxed situation (feelings of love, compassion, and calm) and decreases heart rate. So, the responses are influenced by both mental and physical conditions. Heart rate is measured as beats per minute and a healthy adult would have a Heart rate in the range of 60 to 100 bpm.

Heart rate variability (HRV), an indicator of the balance of these conditions is an estimate of the variations in the time intervals between individual heartbeats. This measure has utility in providing insights into the physiological and psychological state of a person (stress levels, anxiety, etc.).

In the cardiovascular system, the blood pulse propagating throughout the body changes the blood volume in the vessels. Traditional contact methods for measuring HR comprise Electrocardiography (ECG) and Pulse Oximetry (PPG). ECG offers the most accurate HR measurements, but it requires attaching medical electrodes to the subject. Photoplethysmography (PPG) is the process to measure the heart rate with the help of light— a light-emitting diode is shone on the skin (using a wristband) and the sensor analyses the fluctuations in the light level which the skin has absorbed — that is how much light has returned to the sensor. The amount of reflected light changes according to capillary dilation and constriction and correlates with the heart rate.

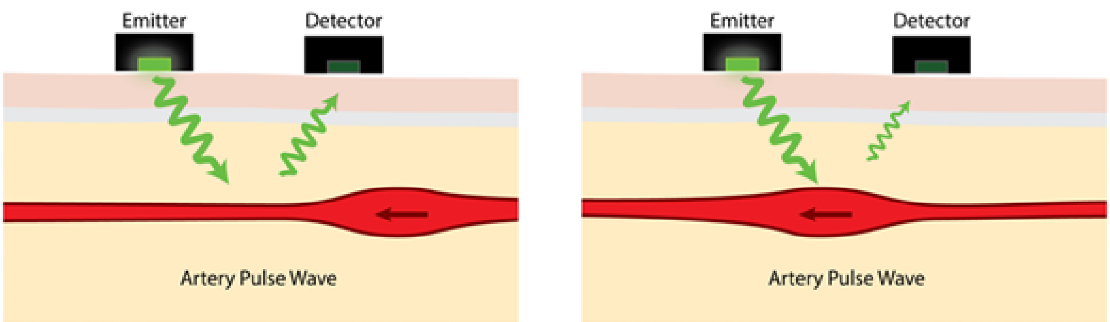

Figure 1.1: Photoplethysmography (PPG) [8]

Thus, this is based on the fact that the optical absorption of hemoglobin varies across the light spectrum, a specific cardiovascular event can be revealed by measuring the color variations of skin reflections. Along with heart rate, other human vital signs such as blood oxygen saturation and related physiological measures can be measured using this technique (PPG). This approach is particularly attractive because it allows tracking of the subject's PPG under free-living conditions, while he or she is performing daily activities, e.g., working on a computer, watching television at home, or in the workplace.

Remote heart rate tracking works on a similar principle but with the use of a webcam which is located at least half a meter away from the subject, unlike a wristband. These remote HR measuring methods are divided into two types: color-based methods, which extract the HR from the slight color changes happening in the skin due to heartbeats, and motion-based methods, which measure the HR from the tiny body motions accompanying the cardiac activity. Thus, this ability to obtain such measurements remotely via a camera/webcam can enable applications outside the medical domain (e.g., affective computing, human-computer interaction), where contact sensors are not feasible.

Recent times(outbreak of the Pandemic COVID-19) has made everyone realize how important it is to keep a track of such Human vitals and many advancements are also made in this field so that any individual can read such Vitals without a need to visit a doctor or without the use of any heavy medical machinery. Though rPPG is in practice for a long time with researches still going on, the tough times have made many young researchers delve more into this topic.

2 Theoretical Background

2.1 Remote Photoplethysmography

Remote photoplethysmography (rPPG) [17] is a color-based method that enables contactless monitoring of human cardiac activities by detecting the pulse-induced subtle color variations on the human skin surface using a multi-wavelength RGB camera. By measuring the variance of red, green, and blue light reflection changes from the skin, as the contrast between specular reflection and diffused reflection.

Specular reflection is the pure light reflection from the skin. Diffused reflection is the reflection that remains from the absorption and scattering in skin tissue, which varies by blood volume changes. Figure 2.1 explains this visually.

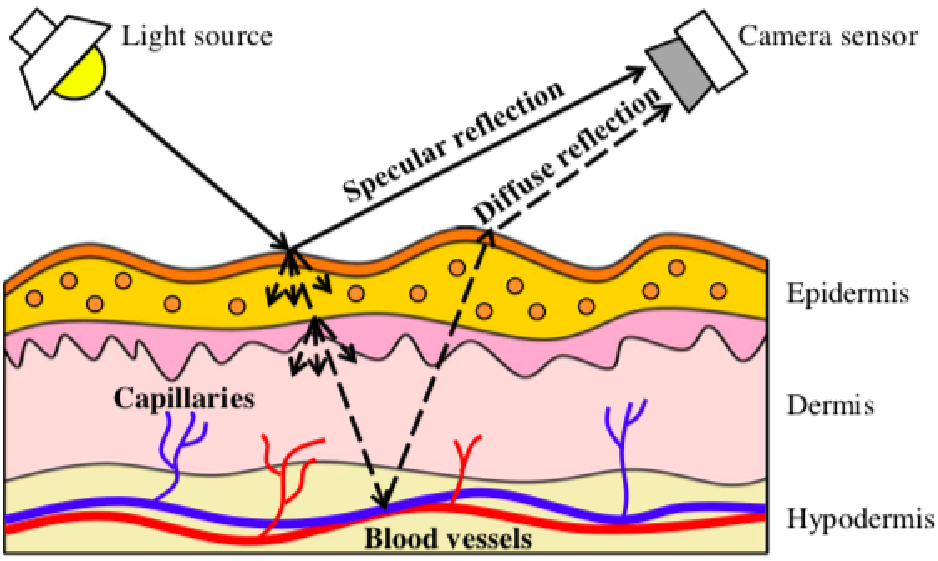

Figure 2.1: Types of Diffusion [8]

The process of rPPG thus involves two steps: detecting and tracking the skin color changes of the subject, and analyzing this signal to compute measures like heart rate, heart rate variability, and respiration rate. Thus, the Combination of face detection, detection of pixel color, and skin reflectivity techniques are building blocks for rPPG.

Recent advances in computer video, signal processing, and machine learning have improved the performances of rPPG techniques significantly. Thus, there would be three different areas of interest to be researched to build an rPPG model and they are mentioned in the next section.

2.2 Open-source Face Tracking Solutions

The first steps of building an rPPG model include detection and tracking of the Human face which can be easily executed by one of the many readily available face-tracking solutions. Some of the famous, open-source face tracking solutions [9] include:

2.2.1 MediaPipe Face Mesh

MediaPipe Face Mesh [13] is a face geometry solution that estimates 468 3D face landmarks in real-time even on mobile devices. It employs machine learning (ML) to infer the 3D surface geometry, requiring only a single camera input without the need for a dedicated depth sensor. Utilizing lightweight model architectures together with GPU acceleration throughout the pipeline, the solution delivers real-time performance-critical for live experiences.

2.2.2 OpenFace

OpenFace is the first toolkit capable of facial landmark detection, head pose estimation, facial action unit recognition, and eye-gaze estimation. OpenFace is similar to Google's FaceNet or Facebook's DeepFace and it is implemented using Python and Torch so it can be run on CPUs or GPUs and it uses dlib and OpenCV libraries. It is capable of real-time performance and can run from a simple webcam without any specialist hardware.

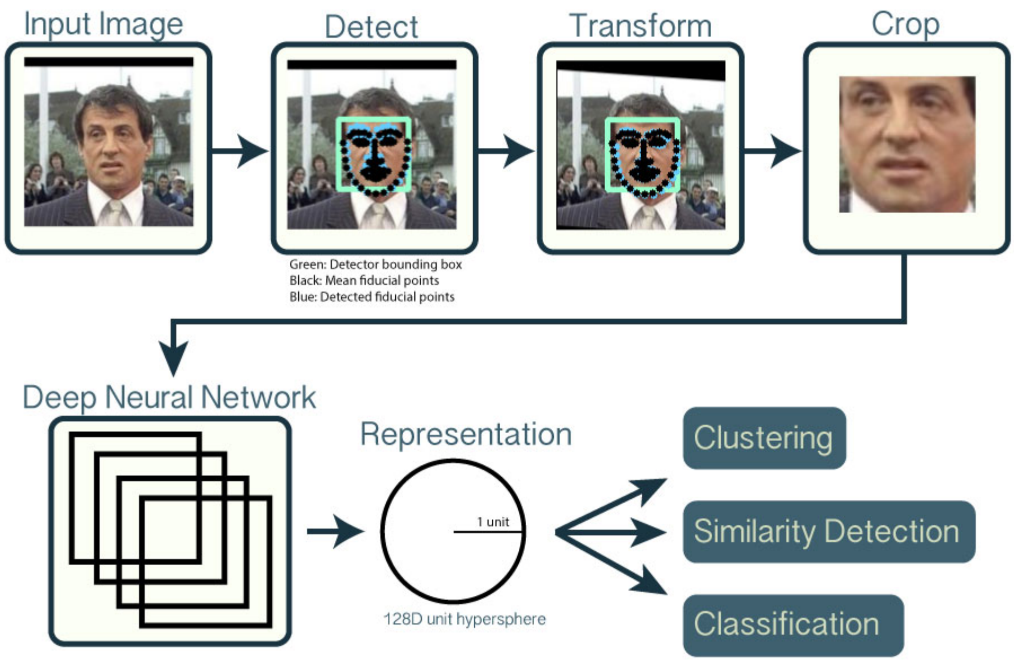

Figure 2.2: OpenFace [9]

2.2.3 Flandmark

Flandmark is an open-source C library (with the interface to MATLAB) implementing a facial landmark detector in static images. Detector parameters learning is written solely in MATLAB and is also part of Flandmark. The input of Flandmark is an image of a face. Each frame is processed separately by Flandmark or Facial Landmark Software.

2.2.4 Kairos API

Kairos face Recognition utilizes the power of computer vision and deep learning to recognize faces. On submitting images and/or videos into the API, the algorithms will analyze the faces found, then the API returns a bunch of useful data about those faces.

2.2.5 Face++ API

- Face Detection - Face Detect API can be used to detect faces within images and get back face bounding box and token for each detected face. You can pass the face token to other APIs for further processing. Detect API also allows you to get back face landmarks and attributes for the top 5 largest detected faces.
- Face Landmark - Face Landmark can be used to get landmarks and attributes by passing its face token to Face Analyze API. You can get a face token by using Detect API. Face Analyze API allows you to process 5 face tokens at a time.

2.2.6 OpenBR

OpenBR face detection software is a communal bio-metrics framework that supports the development of open algorithms and reproducible evaluations. It operates on Windows, Linux, Mac OS X, and Raspbian based operating systems. It implements the 4SF2 algorithm to perform face recognition. The software algorithm also works for age estimation and gender estimation.

2.2.7 Macgyver API

Macgyver API provides a suite of tools around face detection and recognition within images. Capabilities provided include comparing two faces (face recognition), detecting the presence of faces within an image, and returning X, Y coordinates of faces detected in images. This API leverages machine learning and specifically deep convolutional neural networks built in TensorFlow.

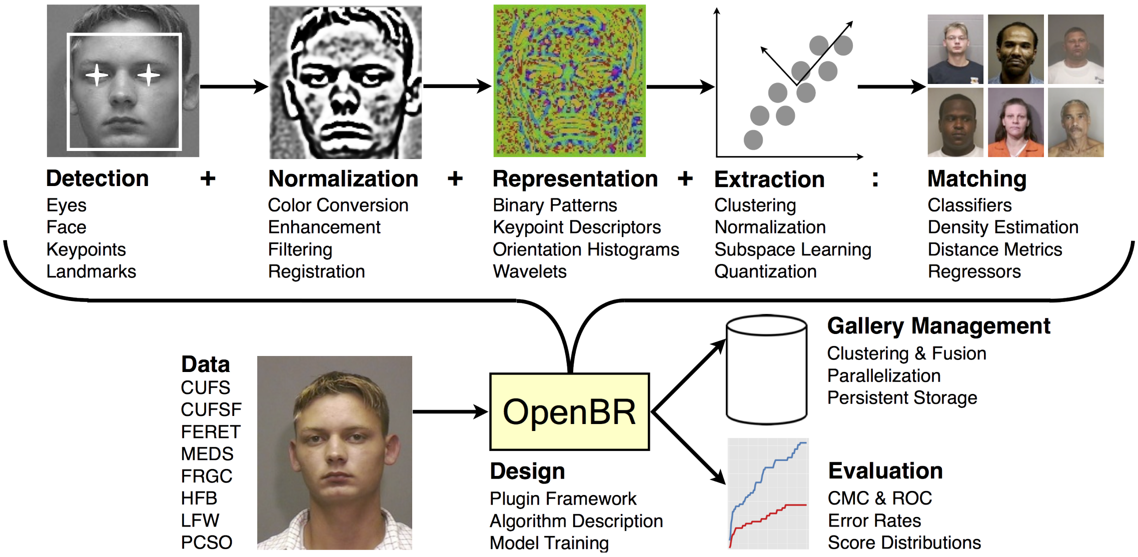

Figure 2.3: OpenBR [9]

2.2.8 OpenFaceTracker

OpenFaceTracker is a facial recognition program capable to detect one or several faces on a picture or a video, and identifying them via a database. The features of this software include real-time processing of images, face identification, the ability to operate on the Windows system, and others.

2.2.9 Rekognition API

Amazon Rekognition detects labels - objects (i.e., people, cars, furniture, clothes, pets), scenes (i.e., woods, beach, a city street) or concepts (outdoors), activities (i.e., playing soccer, skating). It can detect a person in a photo or video, detect facial landmarks, expressed emotion, and save facial metadata. Besides, It can also compare a face in an image with faces detected in another image. (Using RESTFUL API).

Almost all of these solutions are very promising and easy to use in detecting and tracking human faces. A detailed comparison of all these algorithms is made and tabulated the results in the following table 2.1. Based on this table, for this project, I have selected MediaPipe Face detection to detect faces and MediaPipe Face mesh to generate a face mesh with 468 3D face landmarks as it is a very lightweight model, computationally efficient, windows compatible with openly available Python API, and also Face mesh helps better in RoI selection which would be our next step in building rPPG model.

Face Tracking Solution	Openly available	Face Detection	Multi face Detection	Face Mesh(Landmarks)	Lightweight	Python API availability	Windows compatibility	Other details(Age, gender)
MediaPipe Face Mesh	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
OpenFace	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes
Flandmark	Yes	Yes	No	-	Yes	No	-	-
Kairos API	Yes	Yes	Yes	No	No	Yes	Yes	Yes
Face++ API	Yes	Yes	Yes	Yes	No	-	-	Yes
OpenBR	Yes	Yes	No	-	Yes	Yes	Yes	Yes
Macgyver API	Yes	Yes	Yes	-	No	Yes	Yes	-
OpenFaceTracker	Yes	Yes	Yes	No	-	Yes	Yes	-
Rekognition API	Yes	Yes	Yes	Yes	No	No	-	Yes

Table 2.1: Comparison of the Available Face Tracking Solutions

2.3 Region of Interest Selection for Remote Photoplethysmography

After generating a Face mesh around the detected Human face, the next step is to select a Region of Interest (RoI) for further analysis to extract the pulse signal. A recent study by Sungjun Kwon, Jeehoon Kim, Dongseok Lee, and Kwangsuk Park in the paper 'ROI analysis for remote photoplethysmography on facial video' [10] has given me a clear idea on choosing the RoI.

In the camera-based remote PPG (rPPG) monitoring, the region of interest (ROI) is related to the signal quality and the computational load for signal extraction processing. The better quality of the rPPG signal can be extracted from the surface of the body which relatively shows the distinct absorbance variation of blood pulse. The skin whose surface has the fine distribution of capillaries without the hair can be a good candidate for the ROI. The larger size of ROI has a higher chance to extract the better quality of PPG signal, but the computational load is also bigger. Designating the best ROI on the body while minimizing its size is essential for computationally efficient rPPG extraction.

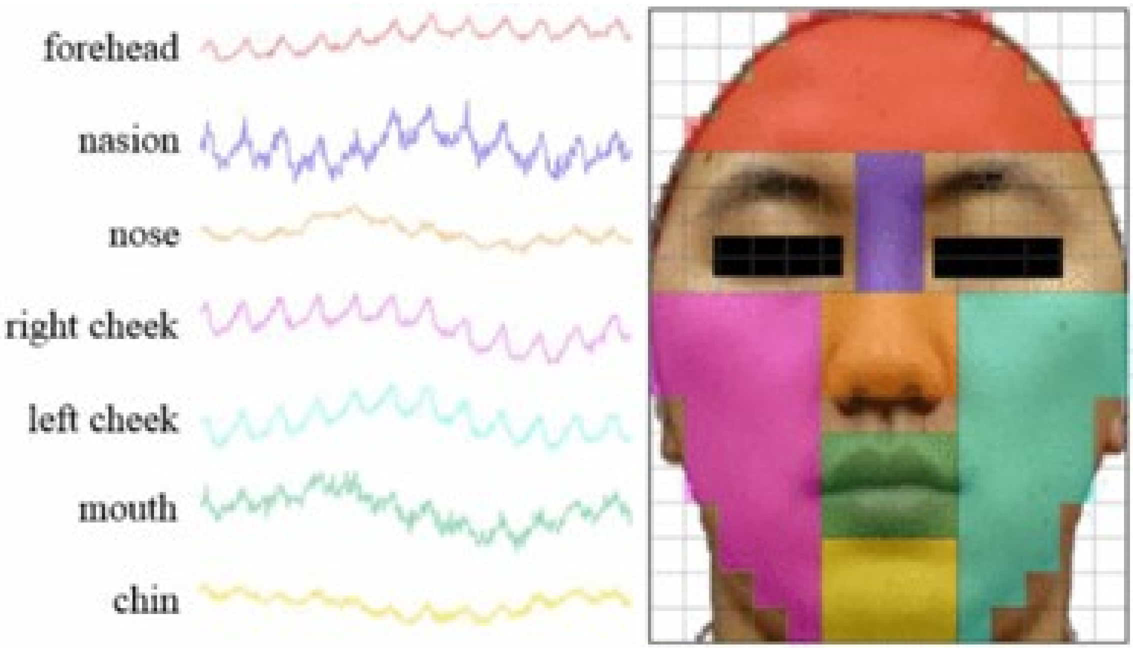

Figure 2.4: ROI - Seven Regions [10]

The face can be divided into seven regions - a forehead, left and right cheeks, a nose, a mouth, a nasion and a chin as shown in Figure 2.4. On evaluating the quality of the signal of each region using the area ratio of high-SNR and high-correlation, and mean and standard deviation (SD) of SNR and correlation coefficient, they found that the forehead and both cheeks especially have a potential to be good candidates for computationally efficient ROI as shown in Figure 2.5. The findings also include that the signal quality from a mouth and a chin was relatively low whereas A nasion and a nose cannot be efficient ROI.

The thick beard can also prevent the monitoring of the variation of absorbance by screening the skin. Because the contracted muscle can disturb the flow of the blood by pressurizing the vessels, the variation also cannot be observed on a video.

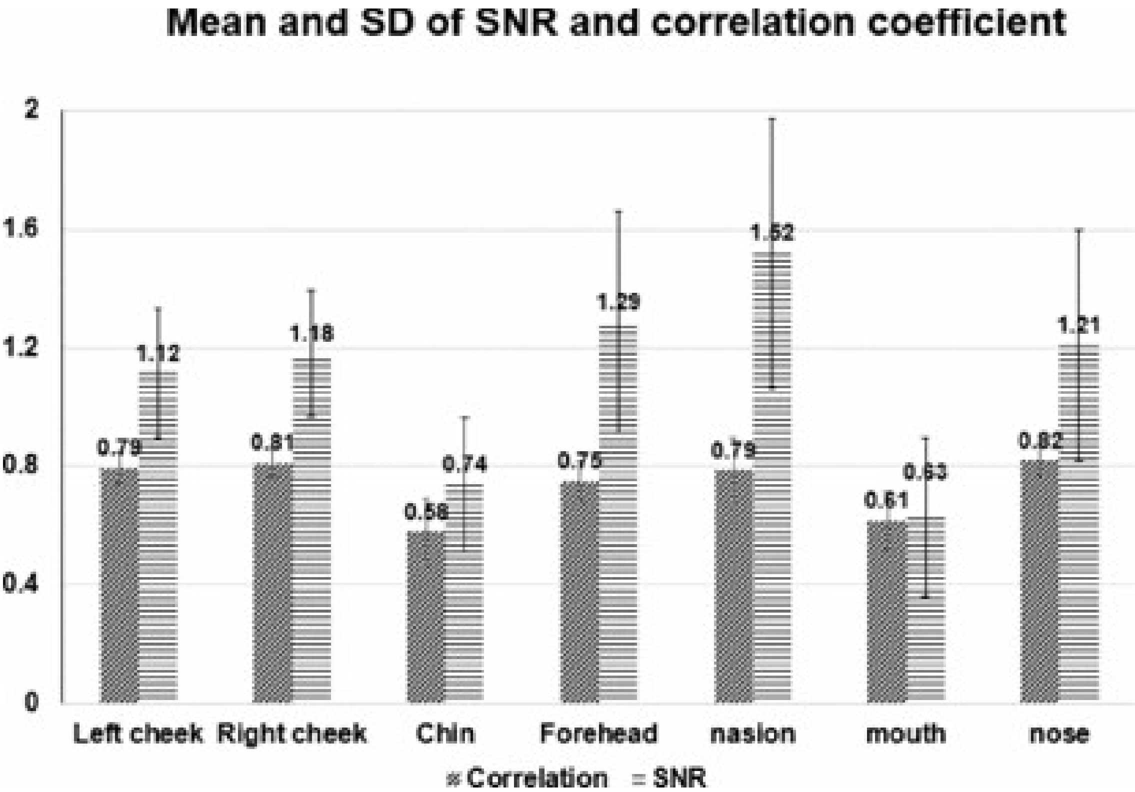

Figure 2.5: ROI - Evaluation [10]

After going through these findings, I have selected two RoIs for this project - Forehead and Cheeks. The model was built and evaluated based on the results obtained by selecting these two RoIs individually and also collectively as a Unit, thus making 3 different combinations

- Forehead,
- Cheeks,
- Forehead and Cheeks

2.4 Overview of Existing Remote Photoplethysmography Methods

The last but most important topic of interest to be researched is the available rPPG methods that were in practice. In recent years, several core rPPG methods have been proposed for extracting the pulse-signal from a video and some of them include:

2.4.1 Blind Source Separation

Blind Source Separation(BSS) [11] is the separation of temporal RGB traces (a set of source signals from a set of mixed signals), without the aid of information (or with very little information) about the source signals or the mixing process using different criteria (PCA Principal Component Analysis/ICA - Independent Component Analysis) [14] into uncorrelated or independent signal sources to retrieve the pulse. PCA and ICA-based methods have comparable results, while PCA reduces computational complexity.

$$Y(t) = W * C(t)$$

Y(t) – Observed signals

W – De-mixing matrix

C(t) – Source signals

Here, we assume that the most periodic signal from Y(t) to be the pulse and so this method is Inefficient with Periodic motion-induced signals.

2.4.2 Chrominance-Based Remote Photoplethysmography

Chrominance-Based rPPG (CHROM) [5] method linearly combines the chrominance signals by assuming a standardized skin tone to white-balance the images. CHROM approach is relatively simple and has been shown to outperform previous baselines such as ICA and PCA. The algorithm first applies a color filter based on a one-class SVM to find skin-colored pixels in each frame of the input sequence. Then, the mean skin color value is projected onto the proposed chrominance subspace, which aims to reveal subtle color variations due to blood flow. The final pulse trace is obtained by bandpass filtering the temporal signals in the XY chrominance colorspace first and then combining the two dimensions of this colorspace into a one-dimensional signal.

This method has better Robustness to subject motion but also has some assumptions related to Light source and Skin-tone.

2.4.3 Using Blood Volume Pulse Vector

Another approach, shortly called as PBV [4] uses the unique signature of the blood volume pulse (PBV-signature) from the color channels of an RGB camera in different wavelengths to explicitly distinguish the pulse-induced color changes from motion noise in RGB measurements.

The weighting vectors derived from the CHROM and PBV methods can be used to guide the component selection of the BSS methods and this method has shown superior accuracy and motion robustness compared to earlier standalone rPPG methods.

2.4.4 Spatial Subspace Rotation

Spatial Subspace Rotation (2SR) [16] considers the subspace of skin pixels in the RGB space and derives the pulse signal by analyzing the rotation angle of the skin color subspace in consecutive frames. To do so, the eigenvectors of the skin pixels correlation matrix are considered. More precisely, the angle between the principal eigenvector and the hyperplane defined by the two others is analyzed across a temporal window.

This algorithm can directly retrieve a reliable pulse signal, and hence no post-processing step (i.e. bandpass filtering) is required. However, this algorithm needs an accurate estimate of the skin color, since it is implicitly assumed that the skin-colored pixels form a single cluster in the RGB space.

2.4.5 Plane Orthogonal to Skin

Plane-orthogonal-to-skin(POS) [17] method defines a plane orthogonal to the skin tone in the temporally normalized RGB space for pulse extraction for removing the specular reflections at the skin surface. Here, we use the knowledge of the blood volume pulse to define a rough projection region on the plane orthogonal to the temporally normalized skin-tone direction and refine an exact projection direction on the plane by real-time tuning.

In detail, given x(t), the POS method goes through three stages. A temporal normalization step is performed before the signal projection on the plane orthogonal to the skin by

$$X_{POS}(t) = x_g(t) - x_b(t)$$

$$Y_{POS}(t) = x_g(t) + x_b(t) - 2x_r(t)$$

Similar to CHROM, the last step is accomplished to tune an exact projection direction within the bounded region defined by the previous step, i.e.

$$y(t) = X_{POS}(t) + \alpha Y_{POS}(t)$$

where α is the same as CHROM. The POS approach is slightly different concerning CHROM, because in the latter the two projected signals are antiphase, while POS directly finds two projection-axes giving in-phase signals. Moreover, to improve the SNR of the signal, the input video sequence is divided into smaller temporal intervals, and pulse rate is estimated from the short video intervals; the final signal is derived by overlap-adding the partial segments.

Algorithm 1 Plane-Orthogonal-to-Skin (POS)

Input: A video sequence containing N frames

Initialize: $\mathbf{H} = \mathbf{zeros}(1, N)$, $l = 32$ (20 fps camera)
for $n = 1, 2, ..., N$ **do**
    $\mathbf{C}(n) = [R(n), G(n), B(n)]^\top \leftarrow$ spatial averaging
    if $m = n - l + 1 > 0$ **then**
        $\mathbf{C^i_n} = \dfrac{\mathbf{C^i_{m \to n}}}{\mu(\mathbf{C^i_{m \to n}})} \leftarrow$ temporal normalization
        $\mathbf{S} = \begin{pmatrix} 0 & 1 & -1 \\ -2 & 1 & 1 \end{pmatrix} \cdot \mathbf{C_n} \leftarrow$ projection
        $\mathbf{h} = \mathbf{S_1} + \dfrac{\sigma(\mathbf{S_1})}{\sigma(\mathbf{S_2})} \cdot \mathbf{S_2} \leftarrow$ tuning
        $\mathbf{H_{m \to n}} = \mathbf{H_{m \to n}} + (\mathbf{h} - \mu(\mathbf{h})) \leftarrow$ overlap-adding
    end if
end for

Output: The pulse-signal $\mathbf{H}$

Figure 2.6: Plane Orthogonal to Skin [17]

Thus, it is based on all of the previously established methods and is more accurate compared to them [1]. Since PBV, CHROM and POS are all approaches to de-mix based on optical/physiological considerations, they share many properties. The bare core algorithm [2] of POS is shown in Figure 2.6. For all these reasons, I have chosen this algorithm for my study on rPPG.

3 Methodology and Conceptual Design

3.1 Structure of the Work

A brief method for extracting heart rate (HR) and heart rate variability (HRV) from the face in real-time using only a consumer-grade webcam and CPU is as follows:

- Region of Interest Masking
- Signal Extraction
- Signal Filtering
- Heart Rate and Heart Rate Variability Calculation

On providing a recorded video or real-time webcam video input, first, the face of the subject is detected and tracked over time. This detected face is then subjected to skin segmentation and pixel information from a selected Region of Interest is collected. Any of the existing rPPG algorithms is used to extract the blood volume pulse(BVP) from the collected pixel information and the extracted BVP is then filtered to get the Heart rate. This process is explained in detail in the next Chapter 'Implementation'. The experimental setup required for the rPPG to estimate Heart rate is explained below:

Experimental setup and Video Recording: The experimental setup [2] consists of a web camera located at least half a meter away from the subject and a synchronized recorder of ECG/PPG attached to the subject. The measurements are performed indoors with the sunlight as the only light source and then a 5 minutes long video sequence is recorded. Though I have used a dataset for studying our rPPG approach, the real-time data recording for Heart rate analysis is performed under such an experimental setup.

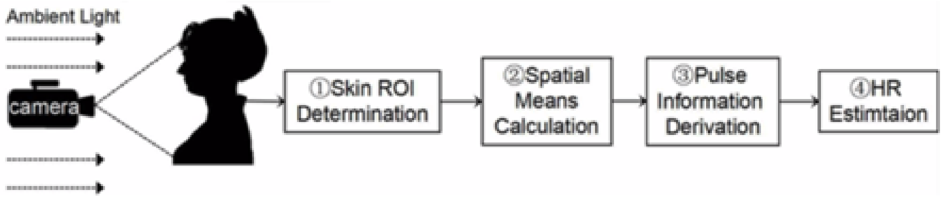

Figure 3.1: Remote Photoplethysmography (rPPG) [14]

3.2 Datasets

3.2.1 Overview of Datasets

A glance at the related literature reveals how there is a lack of a benchmark commonly recognized as suitable for testing rPPG methods. Indeed, experiments are generally conducted either on private datasets or on public ones that are not conceived for rPPG assessment preventing in both cases fair comparisons. Moreover, the different experimental conditions (e.g, illumination, subject movements, in the wild/controlled environment), or different ground-truth reference signals (e.g., electrocardiogram (ECG) or BVP), are likely to prejudice comparisons, too.

But, to study the properties of the proposed rPPG method, the algorithm must be evaluated on an existing dataset conceived for testing the rPPG method. There are a few publicly-available datasets specially designed for the task of remote HR estimation. Stricker et al. firstly released the PURE dataset collected by the camera of a mobile sever robot [15]. Hus et al. released the PFF dataset containing 10 subjects under 8 different situations [7]. In 2018, Xiaobai et al. proposed a dataset designed for HR and heart rate variability (HRV) measurement [12]. Amogh Gudi, Marian Bittner, and Jan van Gemert proposed a novel dataset called 'VicarPPG-2' for the reproducible assessment of rPPG algorithms [2].

A long-debated issue in the pattern recognition field is represented by the bias of the dataset used when performing an analysis. As a matter of fact, running the same algorithm on different datasets may produce markedly different results. In other words, every dataset has its own bias, consequently, the performances reported on a single dataset reflect such biases. rPPG methods make no exception, being defacto very sensitive to different conditions (video compression, different lighting conditions, different setups). But, due to the limited scope, time, and resources of my study, I have selected this VicarPPG-2 dataset as it was publicly available for research purposes, and so I requested via e-mail to the author Mr. Amogh Gudi, who immediately provided the dataset upon proper documentation.

3.2.2 VicarPPG-2 Dataset

Specifically aimed at evaluating rPPG algorithms at estimating heart rate and short-term heart rate variability, this dataset contains 40 different videos recorded in H264 format using a Logitec Brio webcam as well as ECG and PPG Ground Truths. Videos are of 10 participants where each participant was recorded under 4 conditions and each recording session lasted for about 5 minutes. The male to female ratio was 7:3 with an average age of 29±5 years, and skin types ranging on a Fitzpatrick scale from II to IV.

Participants were asked to sit in front of a computer screen (around 1 m distance) on which the instructions were shown, a webcam was mounted on top of the screen and an LED ring lamp was mounted behind the camera. Screen brightness was reduced as far as possible to minimize the influence of screen light on the face.

All videos were recorded using a Logitech Brio webcam at a fixed framerate of 60 fps using an H.264 compliant encoder (Microsoft Media Foundation), and stored in mp4 containers. The recording location was illuminated by natural ambient light in addition to a LED ring lamp (Falcon Eyes DVR-300DVC) to prevent strong shadows and influences of large changes in natural light. The ground truth signals were recorded in the form of synchronized ECG signals at 250 Hz sampling frequency obtained via an Arduino-based ECG board (AD8232), and synchronized PPG signals at 60 Hz obtained via a pulse oximeter device (CMS50E) attached to the left index finger of the participant. The following four scenarios/conditions were recorded for each participant:

- Baseline: Participants sit naturally while watching a relaxing video or reading an article on screen.
- Movement: Participants perform four different types of pre-planned angular body/head movements: turning head side-to-side (shaking), moving the head up and down (nodding), a combination of head-shaking and nodding (round), moving eyes while keeping head still, and naturally bobbing their heads while listening to music (dance).
- Stress: Participants playing a stress-inducing Stroop effect-based game.
- Post-workout: Participants sitting unrestrained after performing fatigue-inducing physical workouts to induce higher heart rates.

Each condition was recorded for a duration of 5 min to allow for the computation of short-term heart rate variability, a total of 200 min of video were collected. Out of 400 collected ground truth files, two were removed from the dataset due to excessive finger movement in the PPG device and gradual detachment of the ECG ground electrode, leading to unusable signals. This dataset stands out as it was explicitly collected for RPPG purposes featuring 5-min long 60 fps camera recordings under various physical/physiological conditions, with simultaneous ECG and PPG ground truth recordings. Also as mentioned earlier that due to the limited scope, time, and resources, my study is limited only to the Baseline videos from this dataset where Participants sit naturally while watching a relaxing video or reading an article on screen and thus keeping the Heart rate within the Nominal Heart rate of an adult human being.

3.3 Evaluation Metrics

To study the generalizability of any rPPG method, we benchmark its performance on a dataset using various metrics. To assess accuracy, we first measure the deviation of the predicted HR measures(using both methods) from the ground truth in terms of error, which is the difference between predicted and true values.

Assuming that the results obtained through the rPPG algorithm and the Cleaned PPG ground truth values to be at the same sampling frequency, for every timestamp, we could compare the peak or non-peak value and come out with the following metrics. The formulae for Accuracy, Precision, Recall, and F1-score are

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall}$$

where

TP = True Positive,

TN = True Negative,

FP = False Positive,

FN = False Negative

Here a 'True Positive' implies a timestamp where the predicted peak matches the original peak value, 'True negative' implies a timestamp where a predicted non-peak matches the original non-peak value. A timestamp is given the value 'False positive' when it is predicted as a peak but is not an original peak and similarly when a true peak is assumed as a non-peak value then it is given the value 'False negative'. Thus, the Detected number of peaks would be a sum of True Positives and False Positives.

4 Implementation

4.1 Region of Interest Masking

The first step in the pipeline is the face detection and fitting of an active appearance model (AAM), which is then used to determine facial landmarks as well as head orientation. The landmarks are used to define a region of interest (RoI) which only contains pixels on the face belonging to the skin. Subsequently, approximately the top two-thirds of the face(forehead, left cheek, and right cheek), where most of the blood vessels are concentrated, is selected as the region of interest (RoI).

4.1.1 Face Detection

Face detection is an important aspect which is used to identify human's face in digital image. This ultimately provides bounding box coordinates defining the subject face.

4.1.2 Face Tracking

After detecting the face in the first frame, the location is used to track the face in the coming frames by only tracking specific features of the face over time.

4.1.3 Adaptive Skin Detection

Skin segmentation is performed on every frame to filter out non-skin pixels by the application of a mask based on RoI.

4.1.4 Choosing Region of Interest

Region of Interest is selected from the skin pixels obtained after the skin segmentation process. Region of Interest plays an important role in rPPG as mentioned under the section 'Region of Interest'.

4.2 Signal Extraction

Within the ROI, the Image is Decomposed into RGB channels and the average of each pixel color (red, green, blue) of the region is computed over time (both specular + diffuse reflections).

Subsequently, the three color signals from R, G, and B channels are combined into a single rPPG signal using the POS method. This method filters out intensity variations by projecting the R, G, and B signals on a plane orthogonal to an empirically determined normalized skin tone vector. The resulting 2-D signal is combined into a 1-D signal via a weighted sum with the weight determined by the ratio of standard deviations of the two signals. This ensures that the resulting rPPG signal contains the maximum amount of the pulsating component.

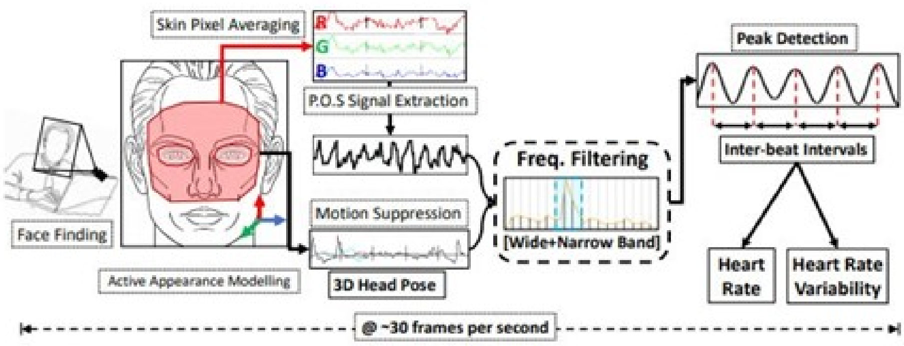

Figure 4.1: Detailed Pipeline of rPPG [3]

The detailed pipeline of the rPPG can be visualized in Figure 4.1

4.3 Signal Filtering

A copy of the extracted rPPG signal is converted to the frequency domain using Fast Fourier Transform. Later, the frequencies outside of the human heart rate range are removed from the spectra using a Band-pass Filter. Two different filters one with narrow band (0.9 – 1.8Hz / 54 – 108 bpm) and another with wide band (0.6 – 2.5 Hz / 36 – 150 bpm) are used in filtering the extracted pulsating signal. These bandwidths have been chosen empirically and depends on the robustness of the subsequent peak detection algorithm to distinguish heat beat peaks from noise (the higher the robustness, the wider this bandwidth can be).

The narrowband is chosen based on the fact that all the recordings of the subjects within the dataset (baseline videos) are performed under normal conditions with no heavy head movements restricting the Heartrate to be around the Nominal range (60 - 100). The wideband is selected to generalize the proposed rPPG method for any real-time Data which may include Heart rates outside the Nominal range. These bandpass filters can either be realized via inverse FFT or a high-order FIR filters.

For our experiment, I have used a 2nd order Bandpass filter with above mentioned passbands.

A welch periodogram of the extracted rPPG signal is plotted with a Hamming window of size 1024 records. This helps in understanding the power spectral density of the signal at different frequencies.

4.4 Heart Rate and Heart Rate Variability Calculation

Once a clean rPPG signal is obtained, we can perform peak detection on it to locate the individual beats in time in the signal. From the located beats, heart rate(HR) and heart rate variability(HRV) can be calculated. There are multiple ways to calculate the HR/HRV from the clean rPPG signal.

4.4.1 Using Inter-Beat Intervals and Calculations

Before calculating HR/HRV, We first extract the inter-beat-intervals (IBIs) [2] from the signal, which are the time intervals between consecutive beats. the extracted inter-beat intervals (IBI) are filtered to remove noise caused by false positive/negative peak detections. First, all IBIs lying outside the range of 250 ms to 2000 ms are excluded (corresponding to the human heart rate range of 30 to 240 bpm). To further remove strong outliers from the signals, intervals farther than three standard deviations from the mean are removed.

Heart rate is calculated by averaging all IBIs over a time window and computing the inverse of it. That is,

$$HR_w = 1/\sqrt{IBI_w}$$

where $\sqrt{IBI_w}$ is the mean of all inter-beat intervals that fall within the time window w. This gives the heart rate in Hertz (assuming IBIs in seconds), and multiplying by 60 gives us the heart rate in beats-per-minute. The choice of this time window can be based on the user's requirement (e.g., instantaneous HR, long-term HR).

Multiple metrics can be computed to express the measure of heart rate variability in different units. We focus on one of the most commonly used time-domain metrics for summarizing HRV called the 'Root mean square of successive differences(RMSSD)', expressed in units of time. As the name suggests, this is computed by calculating the root mean square of the time difference between adjacent IBIs:

$$RMSSD = \sqrt{(1/N)\sum_{i=1}^{N}(IBI_i - IBI_{i+1})^2}$$

where IBIi represents the ith inter-beat interval, and N represents the number of IBIs in the sequence. A graphical example of such HRV calculation is shown in Figure 2. Because RMSSD is more susceptible to noise, only IBIs within the first standard deviation around the mean are considered. Along with RMSSD, we calculate another time-domain HRV metric known as the 'standard deviation of NN intervals(SDNN)', which is simply the standard deviation of all filtered IBIs in the sequence.

4.4.2 Using Welch Periodogram and Filtering

Another simple way of calculating HR is by plotting the Welch Periodogram of the clean rPPG signal to get the power spectral density of the signal distributed over the frequency range and focussing on the frequencies of the human heart rate range (0.9 – 1.8 Hz / 54 – 108 bpm) or (0.6 – 2.5 Hz / 36 – 150 bpm). The frequency with the highest power spectral density could be assumed as the HR. This gives the heart rate in Hertz, and multiplying by 60 gives us the heart rate in beats-per-minute.

4.4.3 Using HeartPy Library

HeartPy is a Python Heart Rate Analysis Toolkit designed to handle (noisy) PPG data collected with either PPG or camera sensors. This is a noise-resistant algorithm that handles PPG data well. It has been implemented in Python and C.

Common measures expressing the HR are the beats per minute (BPM) and the mean inter-beat interval (IBI). HRV is expressed in the median absolute deviation of intervals between heartbeats (MAD), the standard deviation of intervals between heartbeats (SDNN), the root mean square of successive differences between neighboring heartbeat intervals (RMSSD), the standard deviation of successive differences between neighboring heartbeat intervals (SDSD), and the proportion of differences between successive heartbeats greater than 50ms and 20ms (pNN50, pNN20, resp.).

5 Evaluation

5.1 Experiments with the VicarPPG-2 Dataset

Following the process described in the chapter 'Implementation' and using the VicarPPG-2 Dataset (Baseline Videos), rPPG signals were extracted from the recorded videos for different subjects. For this study, I was working on each subject 3 times, once taking only Forehead as RoI, secondly taking Left and Right Cheeks together as RoI, and lastly taking both cheeks and Forehead as RoI. A brief explanation of the pipeline executed for Subject 1 when Both cheeks and Forehead is chosen as RoI is explained below along with figures.

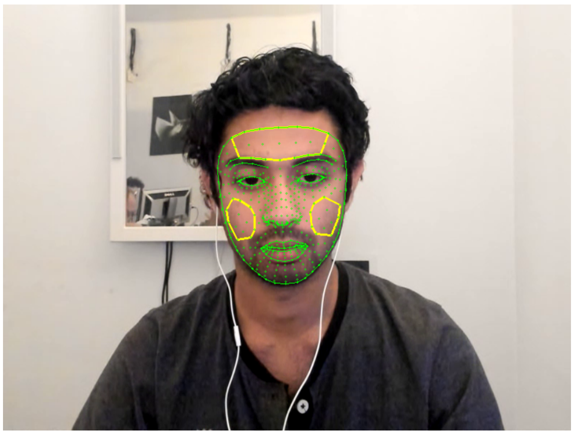

Figure 5.1: Face Detection [2]

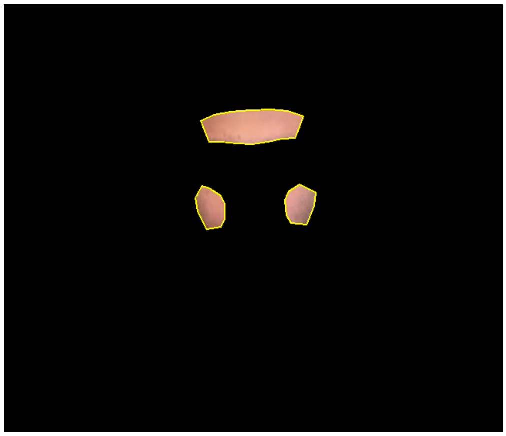

Figure 5.2: Skin Segmentation [2]

As the first step, the Mediapipe library is used to detect the subject's face in a bounding box along with the 428 facial landmarks(Green dots all over the face). As discussed earlier, I have selected Forehead and Cheeks regions as the ROI and so drawn contours around these regions to form RoI (yellow closed contours) assuming most of the blood vessels to be concentrated over these regions. Figure 5.1 shows the Detected face of Subject 1 with RoI marked.

Later skin segmentation is performed on every frame to filter out non-skin pixels by the application of a mask and the resulted output can be seen in Figure 5.2.

Now that the RoI is captured, the next step is to extract the raw signals from these regions. Within this ROI, the Image (single Frame) is Decomposed into RGB channels and the average of each pixel color (red, green, blue) of the region is computed over time(frames) as shown in Figure 5.3

To get a better understanding, the raw signal from the Green channel alone can be visualized in Figure 5.4

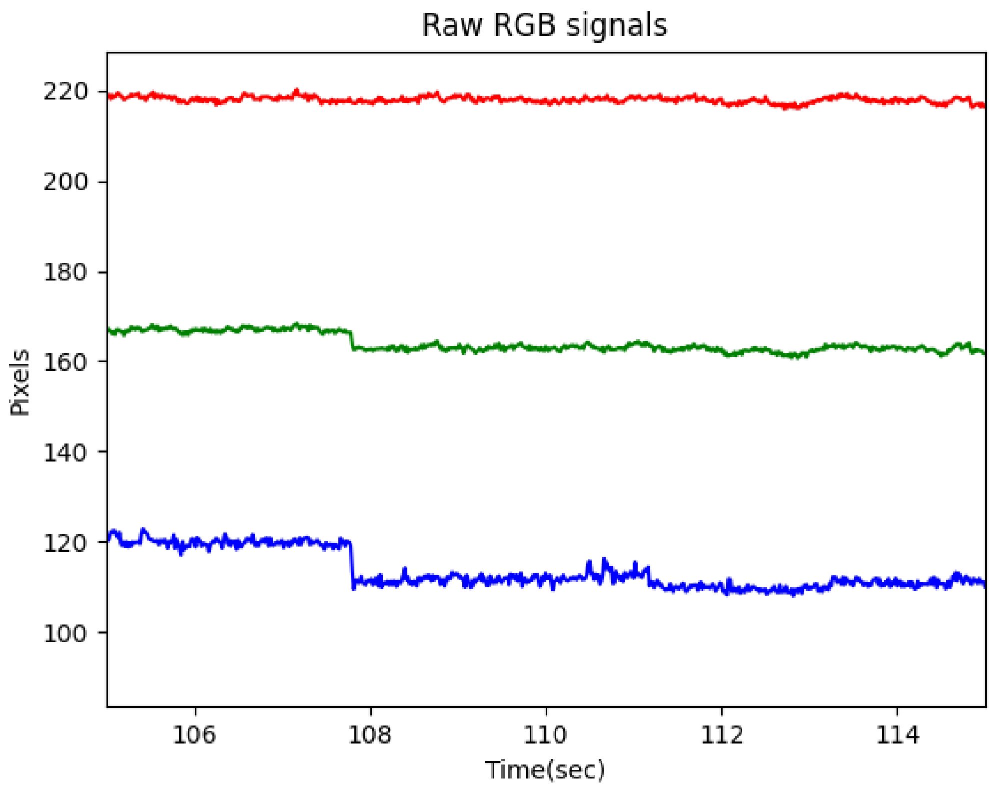

Figure 5.3: Raw RGB Signals from ROI

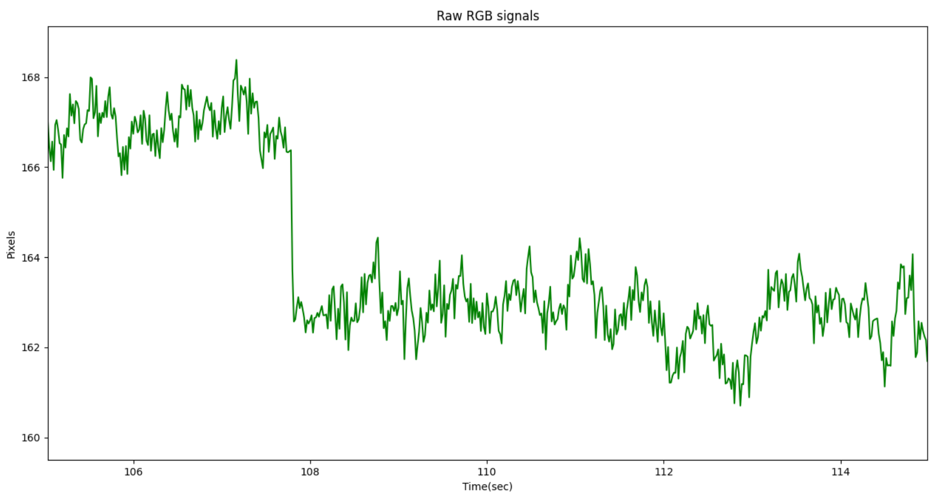

Figure 5.4: Green Channel - Zoomed in

The selected algorithm POS is then applied on these three color signals from R, G, and B channels to combine into a single rPPG signal. Thus, intensity variations are filtered out by projecting the R, G, and B signals on a plane orthogonal to an empirically determined normalized skin tone vector. The resulting 2-D signal

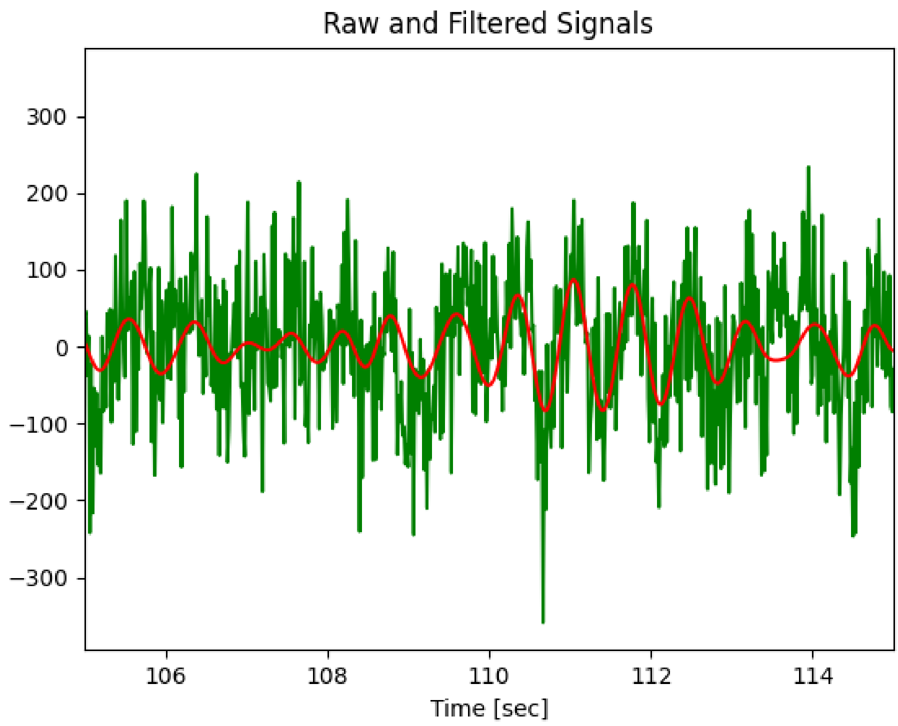

Figure 5.5: Narrow Bandpass Filtered Blood volume pulse

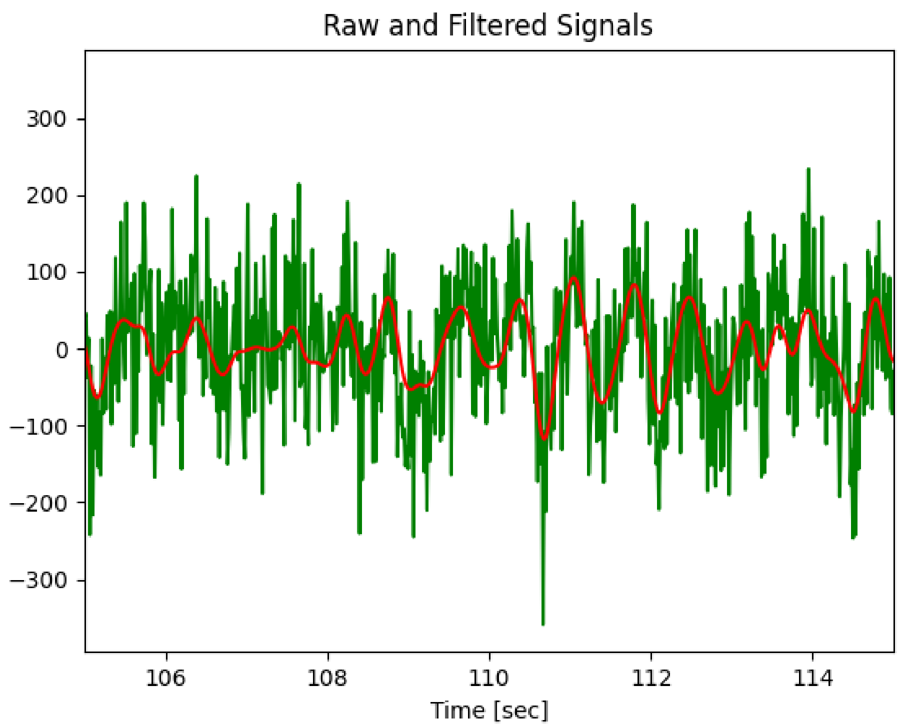

Figure 5.6: Wide Bandpass Filtered Blood volume pulse

is combined into a 1-D signal via a weighted sum with the weight determined by the ratio of standard deviations of the two signals. This ensures that the resulting rPPG signal contains the maximum amount of the pulsating component.

As the obtained pulsating signal contains artifacts due to many disturbances, a Butterworth bandpass filter of 2nd order is used to filter out the frequencies outside of the human heart rate range. Two different filters one with narrowband (0.9 – 1.8 Hz / 54 – 108 bpm) and another with wideband (0.6 – 2.5 Hz / 36 – 150 bpm) are used in filtering the extracted pulsating signal. The filtered and unfiltered extracted pulse signal can be seen in Figures 5.5 and 5.6.

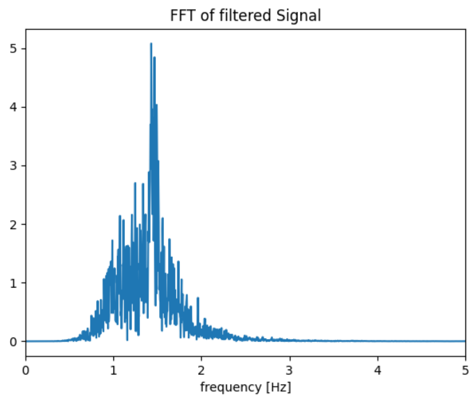

Figure 5.7: FFT plot (0.9 – 1.8 Hz)

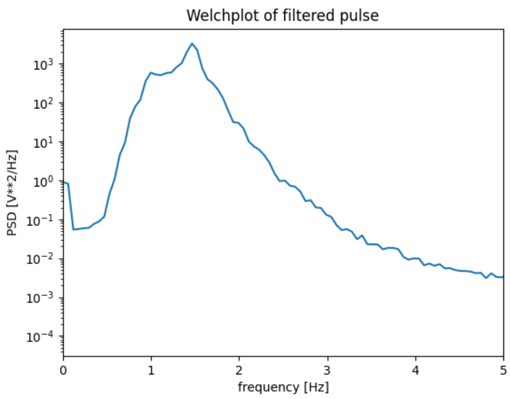

Figure 5.8: Welch plot (0.9 – 1.8 Hz)

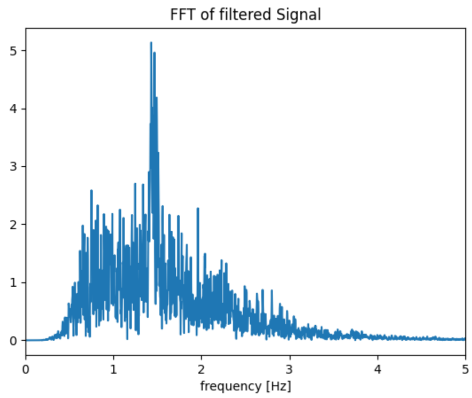

Figure 5.9: FFT plot (0.6 – 2.5 Hz)

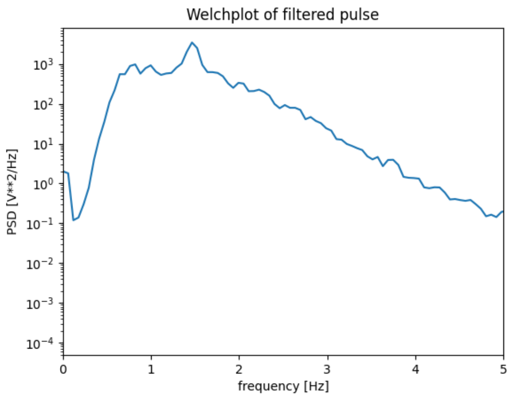

Figure 5.10: Welch plot (0.6 – 2.5 Hz)

This filtered rPPG signal is then converted to the frequency domain using fast Fourier Transform and also a welch periodogram of the filtered rPPG signal is plotted with a Hamming window of size 1024 records to understand the power spectral density of the signal at different frequencies. This can be realized by Figure 5.7, Figure 5.8 for Narrow bandpass filtered BVP and by Figure 5.9, Figure 5.10 for Wide bandpass filtered BVP.

From these figures, it is already vaguely clear that the frequency of around 1.5Hz is having the highest power spectral density for both Narrow and Wide bandpass

Filtering. However, let's continue with the process of calculating the Heart rate from the generated rPPG signal.

The signal that we have now is the clean rPPG signal which could be used further to detect the peaks and to calculate Human Vitals importantly the Heart rate. Out of all the available methods to calculate the Heart rate as mentioned under the section 'Output Calculation', I have used 'Using Welch Periodogram and Filtering' and 'Using HeartPy Library' to analyze the algorithm.

Using Welch Periodogram and Filtering, the first step is to get the frequency with the highest power spectral density. For Narrow bandpass Filtering, we concentrate only on the frequency range that comprises the Nominal human heart rate (0.9 – 1.8 Hz / 54 – 108 bpm) and found out that the frequency of around 1,465Hz is having the highest PSD. For Wide bandpass Filtering, we concentrate only on the frequency range (0.6 – 2.5 Hz / 36 – 150 bpm) and found out that the same frequency of around 1,465Hz is having the highest PSD. So, now multiplying this value with 60 (to convert from bps to bpm), we get the Heart rate of the subject in Beats per minute. Thus, the calculated Heart rate of this subject when Forehead and both Cheeks are taken as RoI is 87.89 bpm.

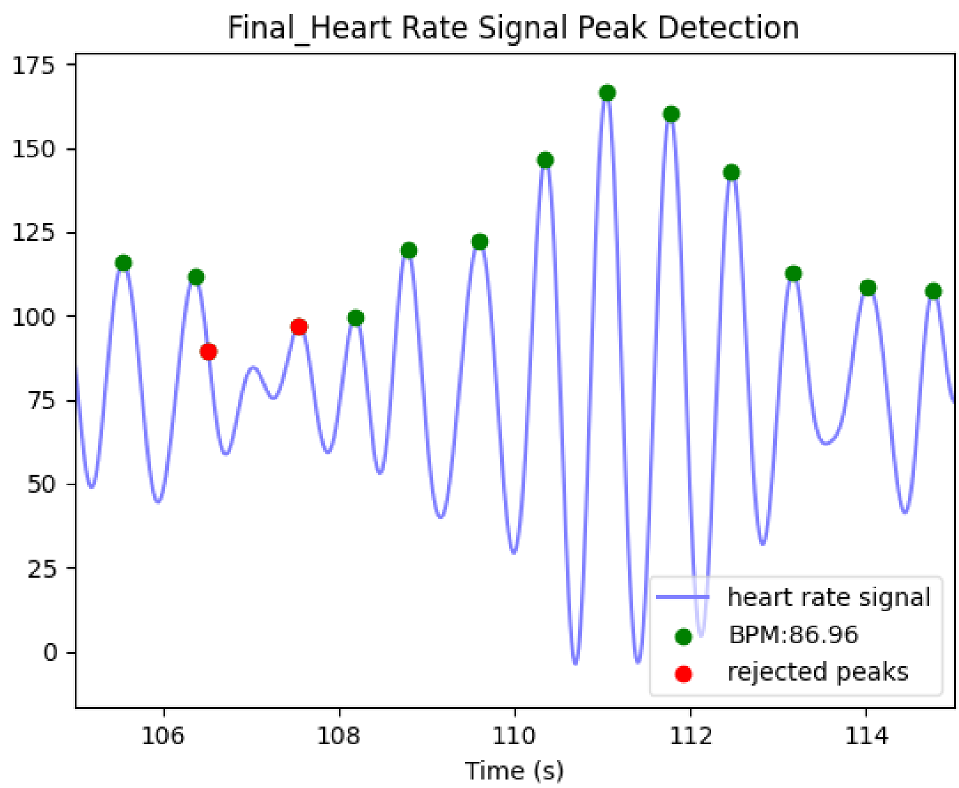

Figure 5.11: HeartPy - Signal Peak detection for Narrow BandPass filtered

As it can be noticed, this may not be the average Heart rate of the subject during that time sequence but the most repeated one as it is extracted based on the highest PSD from the welch plot. So to get a second opinion and to recheck the algorithm, I have employed the HeartPy library to calculate the Heart rate again. After feeding in the clean rPPG signal and the sampling frequency of the recording, the HeartPy Library returns a lot of information about the Subject which includes Vital signs such as HR/HRV and Respiration rate. Figure 5.11 and 5.12 shows the generated pulse using the HeartPy library which includes Detected peaks, rejected peaks, and the calculated Heart rate for Narrow and Wide bandpass filtering respectively.

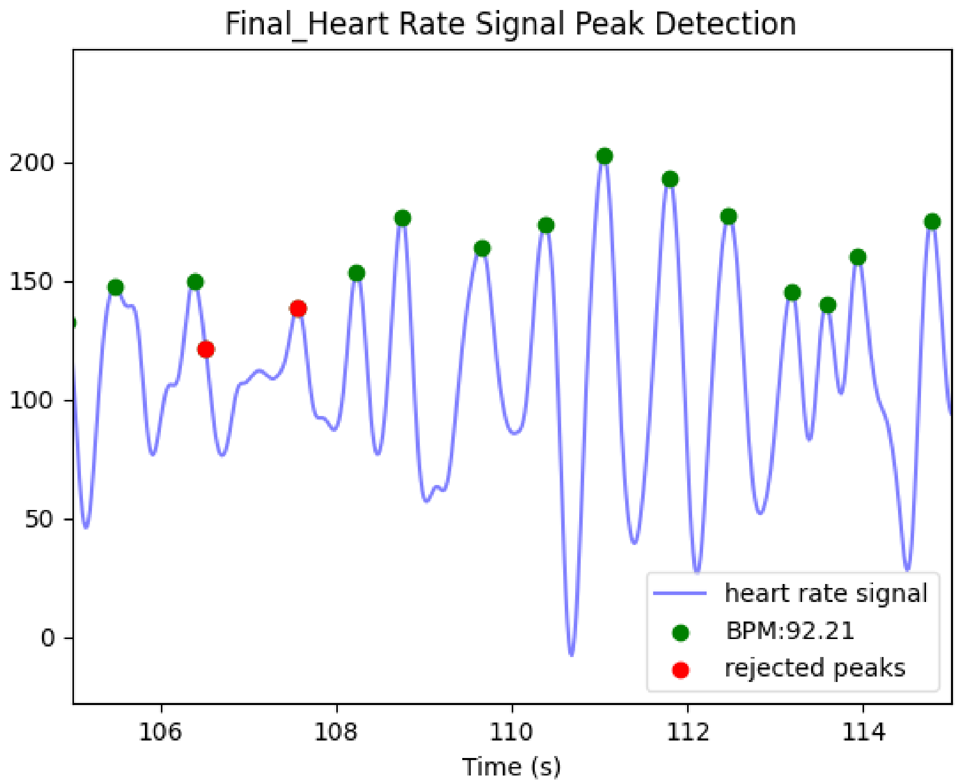

Figure 5.12: HeartPy - Signal Peak detection for Wide BandPass filtered

The Detected Heart rate through this algorithm is around 86,96 bpm and the Respiration rate is 16 breaths per minute when a Narrow bandpass filter is employed , where as the Detected Heart rate is 92,21 bpm and the Respiration rate is 14 breaths per minute when a Wide bandpass filter is employed. The normal respiration rate for an adult at rest is 12 to 20 breaths per minute. A respiration rate under 12 or over 25 breaths per minute while resting is considered abnormal. Also, the detected IBI(Inter Beat Interval) is 689.99 ms and 650.65 ms for Narrow and Wide bandpass filtering respectively.

Similarly, the Heart rate and other important vitals are calculated for Subject 1 by choosing only Forehead or only Cheeks as the RoI. This process is repeated for all the subjects in the Dataset and the results are noted down.

As we are working with the Dataset, we are already provided with the Ground Truth values of Heart rates for all the subjects which were recorded in the form of synchronized ECG signals at 250 Hz sampling frequency obtained via an Arduino based ECG board (AD8232), and synchronized PPG signals at 60 Hz obtained via a pulse oximeter device (CMS50E) attached to the left index finger of the participant. The PPG signals are further manually cleaned to synchronize between video and ground truth values. Thus, these Cleaned PPG signals could be used to find the original Heart rate of the subjects at the time of recording and compare with the one that we have calculated to get the accuracy of the rPPG algorithm we developed.

In a similar manner to what we did to the filtered rPPG signal, for comparison, this ground truth rPPG signal can also be converted to the frequency domain using Fast Fourier Transform, and also a welch periodogram of this signal could be plotted with a Hamming window of size 1024 records to understand the power spectral density of this signal at different frequencies. The Welch plot of the original/ground truth values is shown in Figure 5.13

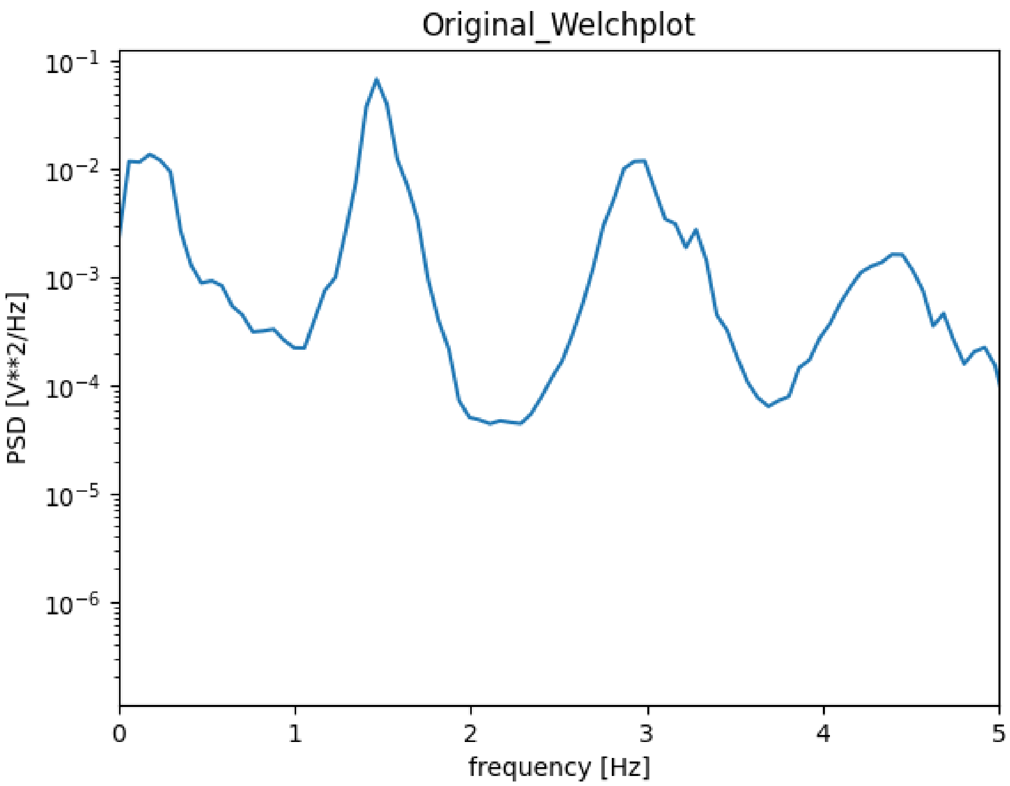

Figure 5.13: Welch plot - Cleaned PPG signal

Now to calculate the Heart rate from these cleaned PPG signals, I have used the first method mentioned under the section 'Output Calculation' called 'Using Inter-Beat Intervals and Calculations'. As the Ground truth values also contain the information of peaks, no further peaks correction needs to be performed while calculating the inter-beat-intervals(IBI) and Heart rate can be then calculated using the formula

$$HR_w = 1/\sqrt{IBI_w}$$

This gives the heart rate in Hertz (assuming IBIs in seconds), and multiplying by 60 gives us the heart rate in beats-per-minute. For subject 1, the calculated Heart rate from the Ground Truth values is 87.97 bpm (From figure 5.13, Highest PSD is around 1.467Hz frequency in the Welch plot).

Now that we have calculated the Heart rate using both Detected pulse signals from rPPG method and also from the ground truth values, let's analyze the method by calculating the evaluation metrics as mentioned under the section 'Evaluation Metrics'. For Subject 1 when Forehead and both cheeks are chosen as RoI, we got the results that are quite promising with Errors being around 0.08 for HR calculated using Welch Plot and 1.01 for HR calculated using HeartPy library when a narrow bandpass filter is used. The accuracy we obtained here is nearly 99 percentage with 98 percentage precision, 82 percentage recall, and 89 percentage f1 score. When a wide bandpass filter is used, the Errors are around 0.08 for HR calculated using Welch Plot and 4.24 for HR calculated using HeartPy library with an accuracy of 99 percentage, 92 percentage precision, 76 percentage recall, and 84 percentage f1 score.

This process is repeated using both the filter passbands for ground truth values of all the subjects and the results are noted down. This helps in the evaluation of the proposed rPPG method. An in-depth analysis and evaluation of the proposed rPPG method based on results obtained with the help of the VicarPPG-2 dataset is explained below based on the results tabulated in Table 5.1 and 5.2 when Narrow and Wide bandpass filters are used respectively.

The results are quite promising with an average accuracy of around 98.7 percentage in both the filter passbands. Also, the precision, recall, and f1 score always came out to be in the range of 60-70 percent on average, these lower values are expected due to the signal inaccuracies that occur between PPG and rPPG signal due to many known factors. As our rPPG method is working well with the dataset, now we can test it directly with the real-time data where we have no information about the ground truth values and we estimate the Heart Rate of a person.

		Number of peaks			Heartrate						Metrics			
Subject	**ROI**	**Original**	**Detected**	**Original (bpm)**	**Welch Plot (bpm)**	**Error**	**HeartPy (bpm)**	**Error**	**IBI (ms)**	**Breathing Rate (bpm)**	**Accuracy**	**Precision**	**Recall**	**f1_score**
	Forehead		380		87,89	0,08	86,58	1,39	692,97	16	0,99	0,97	0,77	0,86
1	Cheeks	478	410	87,97	87,89	0,08	87,65	0,32	684,55	14	0,99	0,97	0,83	0,89
	Forehead + Cheeks		403		87,89	0,08	86,96	1,01	689,99	16	0,99	0,98	0,82	0,89
	Forehead		364		66,79	0,08	68,37	-1,5	877,62	15,5	0,99	0,75	0,75	0,75
2	Cheeks	365	362	66,87	66,79	0,08	68,43	-1,56	876,73	15,6	0,99	0,79	0,79	0,79
	Forehead + Cheeks		360		66,79	0,08	67,81	-0,94	884,8	17,39	0,99	0,78	0,77	0,77
	Forehead		439		77,34	1,5	79,84	-1	751,52	20,06	0,99	0,72	0,72	0,72
3	Cheeks	438	436	78,84	80,86	-2,02	79,79	-0,95	751,96	16,55	0,99	0,7	0,7	0,7
	Forehead + Cheeks		439		80,86	-2,02	79,84	-1	751,52	20,06	0,99	0,7	0,7	0,7
	Forehead		341		63,28	-0,8	65,24	-2,76	919,69	20	0,98	0,52	0,5	0,51
4	Cheeks	352	347	62,48	63,28	-0,8	68,99	-6,51	869,7	8	0,98	0,49	0,48	0,48
	Forehead + Cheeks		348		63,28	-0,8	65,49	-3,01	916,19	12	0,98	0,52	0,52	0,52
	Forehead		303		56,25	4,39	70,13	-9,49	855,55	8	0,98	0,55	0,5	0,52
5	Cheeks	333	320	60,64	56,25	4,39	71,03	-10,39	844,65	12	0,98	0,55	0,52	0,54
	Forehead + Cheeks		303		56,25	4,39	66,3	-5,66	904,99	10	0,98	0,54	0,49	0,51
	Forehead		422		73,83	1,66	76,08	-0,59	788,61	9,19	0,99	0,73	0,71	0,72
6	Cheeks	428	405	75,49	73,83	1,66	76,69	-1,2	782,32	10	0,99	0,75	0,71	0,73
	Forehead + Cheeks		423		73,83	1,66	76,21	-0,72	787,31	9,19	0,99	0,74	0,73	0,74
	Forehead		365		70,31	-2,18	72,82	-4,69	823,97	14	0,99	0,69	0,67	0,68
7	Cheeks	377	353	68,13	70,31	-2,18	71,62	-3,49	837,81	14	0,99	0,71	0,66	0,68
	Forehead + Cheeks		363		70,31	-2,18	71,69	-3,56	836,9	14	0,99	0,72	0,69	0,7
	Forehead		362		66,8	2,37	70,14	-0,97	855,46	7,93	0,98	0,41	0,4	0,41
8	Cheeks	366	365	69,17	66,8	2,37	70,11	-0,94	855,74	9,82	0,98	0,4	0,4	0,4
	Forehead + Cheeks		365		66,8	2,37	69,96	-0,79	857,69	7,73	0,98	0,42	0,42	0,42
	Forehead		400		77,34	-1,01	77,94	-1,61	769,86	14	0,99	0,76	0,73	0,74
9	Cheeks	418	409	76,33	77,34	-1,01	77,84	-1,51	770,79	13,77	0,99	0,78	0,76	0,77
	Forehead + Cheeks		411		77,34	-1,01	77,69	-1,36	772,29	17,62	0,99	0,78	0,77	0,78
	Forehead		439		80,86	-1,11	79,77	-0,02	752,19	15,27	0,99	0,7	0,65	0,68
10	Cheeks	469	439	79,75	80,86	-1,11	80,14	-0,39	748,72	13,43	0,99	0,77	0,72	0,74
	Forehead + Cheeks		440		80,86	-1,11	79,97	-0,22	750,28	11,35	0,99	0,74	0,7	0,72
											0,987	0,68766667	0,65267	0,6686667

Table 5.1: Evaluation - Results Obtained Using Narrow Bandpass Filter

		Number of peaks		Heartrate							Metrics			
Subject	**ROI**	**Original**	**Detected**	**Original (bpm)**	**Welch Plot (bpm)**	**Error**	**HeartPy (bpm)**	**Error**	**IBI (ms)**	**Breathing Rate (bpm)**	**Accuracy**	**Precision**	**Recall**	**f1_score**
	Forehead		390		87,89	0,08	95,85	-7,88	625,91	14	0,99	0,91	0,75	0,82
1	Cheeks	478	390	87,97	87,89	0,08	92,02	-4,05	652,01	14	0,99	0,92	0,75	0,83
	Forehead + Cheeks		396		87,89	0,08	92,21	-4,24	650,65	14	0,99	0,92	0,76	0,84
	Forehead		344		66,79	0,08	68,32	-1,45	878,15	16	0,99	0,72	0,68	0,7
2	Cheeks	365	344	66,87	66,79	0,08	72,29	-5,42	830	16	0,99	0,75	0,71	0,73
	Forehead + Cheeks		347		66,79	0,08	68,71	-1,84	873,14	16	0,99	0,74	0,7	0,72
	Forehead		435		77,34	1,5	79,83	-0,99	751,58	20,34	0,99	0,74	0,73	0,74
3	Cheeks	438	433	78,84	80,86	-2,02	80,06	-1,22	749,4	19,11	0,99	0,73	0,73	0,73
	Forehead + Cheeks		437		80,86	-2,02	79,81	-0,97	751,8	20,19	0,99	0,74	0,74	0,74
	Forehead		318		63,28	-0,8	68,39	-5,91	877,22	12	0,98	0,52	0,47	0,49
4	Cheeks	352	316	62,48	63,28	-0,8	81,8	-19,32	733,5	10	0,98	0,48	0,43	0,46
	Forehead + Cheeks		326		63,28	-0,8	67,61	-5,13	887,46	18	0,98	0,52	0,48	0,5
	Forehead		262		56,25	4,39	76	-15,36	789,41	14	0,98	0,53	0,42	0,47
5	Cheeks	333	281	60,64	56,25	4,39	81,32	-20,68	737,83	10	0,98	0,49	0,41	0,45
	Forehead + Cheeks		269		56,25	4,39	71,57	-10,93	838,36	14	0,98	0,55	0,44	0,49
	Forehead		414		73,83	1,66	76,65	-1,16	782,74	7,75	0,99	0,72	0,69	0,71
6	Cheeks	428	383	75,49	73,83	1,66	80,24	-4,75	747,71	10	0,99	0,75	0,67	0,71
	Forehead + Cheeks		414		73,83	1,66	76,39	-0,9	785,43	9,62	0,99	0,74	0,72	0,73
	Forehead		331		70,31	-2,18	80,5	-12,37	745,3	14	0,99	0,67	0,59	0,62
7	Cheeks	377	333	68,13	70,31	-2,18	78,51	-10,38	764,23	14	0,99	0,67	0,59	0,63
	Forehead + Cheeks		318		70,31	-2,18	76,4	-8,27	785,34	10	0,99	0,71	0,6	0,65
	Forehead		354		66,8	2,37	70,98	-1,81	845,32	10	0,98	0,4	0,39	0,4
8	Cheeks	366	349	69,17	66,8	2,37	70,32	-1,15	853,28	18	0,98	0,4	0,38	0,39
	Forehead + Cheeks		352		66,8	2,37	70,49	-1,32	851,2	8	0,98	0,4	0,39	0,39
	Forehead		365		77,34	-1,01	81,21	-4,88	738,87	16	0,99	0,75	0,66	0,7
9	Cheeks	418	376	76,33	77,34	-1,01	83,03	-6,7	722,59	18	0,99	0,74	0,67	0,71
	Forehead + Cheeks		353		77,34	-1,01	78,61	-2,28	763,28	14	0,99	0,78	0,66	0,71
	Forehead		393		80,86	-1,11	81,22	-1,47	738,71	16	0,99	0,69	0,58	0,63
10	Cheeks	469	405	79,75	80,86	-1,11	85,36	-5,61	702,94	16	0,99	0,71	0,61	0,65
	Forehead + Cheeks		403		80,86	-1,11	81,1	-1,35	739,83	16	0,99	0,73	0,63	0,67
											0,987	0,67066667	0,601	0,6336667

Table 5.2: Evaluation - Results Obtained Using Wide Bandpass Filter

5.2 Experiments with Real-time Webcam Data

We have seen that the algorithm is producing results that are quite promising with very near accuracy and now the same algorithm could be used for real-time data with no prior knowledge of ground truth values.

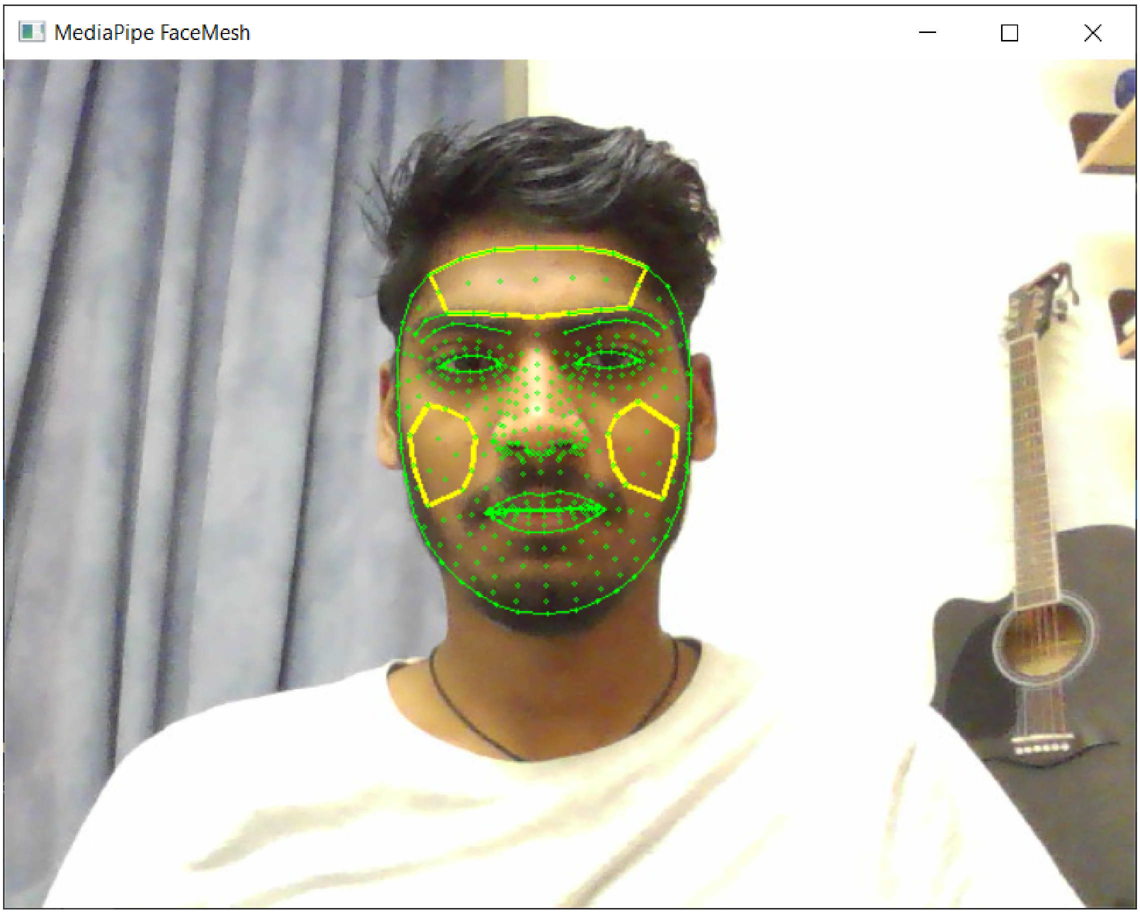

Figure 5.14: Real-time Face Tracking

The experimental setup is almost similar to that of the Dataset videos recording but the ECG or PPG recorders are not attached to the subject to get the ground truth values. A video sequence of 2 min is recorded for the subject and the video is processed through the pipeline as mentioned above to extract the Heart rate and other Vital signs. Figure 5.14 shows the subject along with the ROI marked on the detected face.

The results are tabulated in Table 5.3 where it says that the Detected Heart rate to be around 75-80 bpm when a narrow bandpass filter is used whereas it is 80-95 when a wide bandpass filter is used. This is due to the fact the motion and other artifacts play their role as they are not filtered out when a wide passband is used. Figure 5.15 shows the generated pulse using the HeartPy library which includes Detected peaks, rejected peaks, and the calculated Heart rate.

Testing conditions	Region of Interest	Number of Peaks	Heart Rate Welch Plot (bpm)	HeartPy (bpm)	Inter-beat Intervals
KG (Narrow)	Forehead	138	80,86	80,8	742,53
	Cheeks	141	75,56	81,1	737
	Forehead + Cheeks	147	73,83	79,88	751,15
			76,75	80,593	
KG (Wide)	Forehead	153	79,1	94,48	635,32
	Cheeks	148	74,31	97,21	615,86
	Forehead + Cheeks	150	84,38	94,52	634,8
			79,263333	95,403	

Table 5.3: Heart Rate of Real-time Data

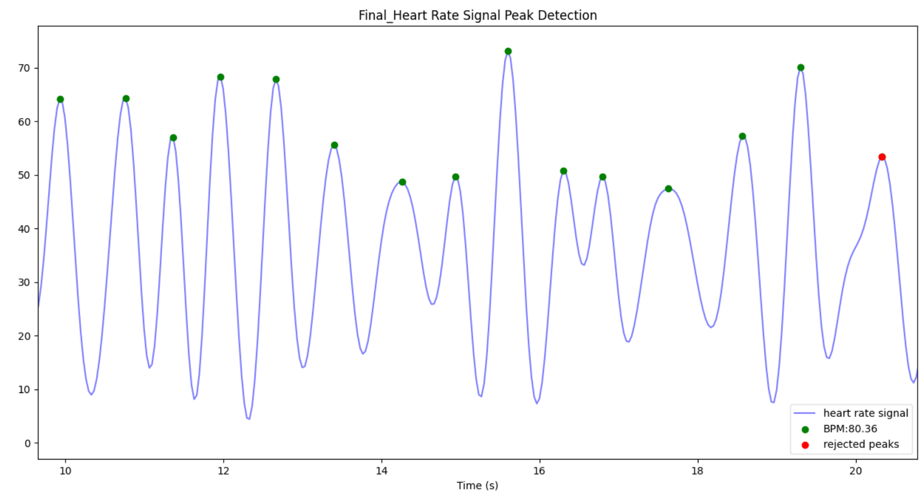

Figure 5.15: Detected Heart Rate

5.3 Results and Discussion

The aim of this project was to study and build an rPPG model to detect Heart rate from the extracted blood volume pulse signal. The results as shown in Figure 5.1 give us a clear idea about how close we are to the aimed target. Almost for all the subjects, the Predicted Heart rate is almost near to the Original Heart rate measured by the PPG device.

But as we are not suppressing the motion artifacts explicitly in our method, the results are much better only when a narrow bandpass filtering is employed but this restricts human Heart rate detection within a limited range like seen above in the section 'Experiments with Real-time Webcam Data'. When a subject is in relaxing condition with not many head motions while recording the video, the method works spectacular with very high accurate results estimating Heart rate that is very very close to the original Heart rate as a narrow passband could be employed in such a condition. Whereas the other way results are accurate enough to get an approximate range of Heart rate estimation for that subject under given conditions.

6 Conclusions

6.1 Summary

Pulse rate estimation using remote photoplethysmography(rPPG) [6] is an ongoing and growing research area. In many respects, it is also a mature discipline encompassing a remarkable amount of results both in terms of algorithmic principles introduced and also concerning the acquired knowledge over time. However, the main agenda of this study is to gain experience in this field and to build an rPPG model from scratch. Thus deep research was done on many important topics needed for the algorithm implementation.

In this contribution, a state-of-the-art rPPG algorithm called POS is selected and evaluated. For this purpose, a new, publicly available dataset VicarPPG-2 containing 10 subjects captured under similar illumination conditions has been introduced. A thorough experimental evaluation of the selected approach has been conducted using the dataset and assessed performance in a principled and unbiased way.

Pre-Processing: Skin pixels play a significant role in the extraction of rPPG signal, therefore, thorough research was done for the region of interest (ROI) selection and face tracking. The model segments skin pixels from non-skin very accurately.

rPPG Signal Extraction: After detection and tracking ROI for signal extraction, the next step was computing the spatial red, green and blue channel mean of skin segmented pixels to minimize camera quantization error. Averaged values of the RGB channels are then temporally normalized and projected to plane orthogonal to skin-tone. The projected signal is alpha tuned to extract the signal.

Post-processing: Estimation of heart rate was done by computing power spectral density PSD, applying Fast Fourier transformation (FFT) on rPPG signal. It is then bandpass filtered to analyze only frequencies of interest. The maximum power spectrum represents the frequency of instant heart rate. Heart rate and other Vital signs are also extracted using HeartPy Library.

Obtained results show that the rPPG algorithm has a stable behavior, but overall it has been noticed that performance is highly dependent on a careful optimization of parameters. But in fact, the accuracy of this method for heart rate analysis is in the same range or beyond several fully supervised deep learning methods, albeit without any rPPG specific training or fine-tuning. It also surpasses all existing methods for heart rate variability estimation and sets some of the first benchmarks for heart rate variability analysis on this dataset.

6.2 Future Work

Due to the limited time and resources, the scope of the project was limited and couldn't be expanded further. But if provided time, further study can be done in various aspects such as improving the algorithm by eliminating the motion artifacts or illumination artifacts, working on a full set of videos from the VicarPPG-2 dataset where subjects were recorded under different conditions, working with other publicly available or self-recorded datasets, or studying the algorithm under different physical/environmental conditions, etc.,

List of Figures

List of Tables

Acknowledgements

I would like to thank Mr. Amogh Gudi from the Vicarious Perception Technologies (VicarVision) at the Delft University of Technology for his support in providing the VicarPPG-2 dataset.

I would also like to thank Mr. Florian Wolling from the Ubiquitous Computing department at the University of Siegen for his continuous support during the project.

Bibliography

[1] Giuseppe Boccignone et al. „An Open Framework for Remote-PPG Methods and their Assessment". In: *IEEE Access* 8 (2020). Pp. 216083–216103.

[2] Amogh Gudi; Marian Bittner; Jan van Gemert „Real-Time Webcam Heart-Rate and Variability Estimation with Clean Ground Truth for Evaluation". In: *Applied Sciences* 10 (2020), 23.

[3] Amogh Gudi et al. „Efficient Real-Time Camera Based Estimation of Heart Rate and Its Variability". In: Oct. 2019, pp. 1570–1579. DOI: `10.1109/ICCVW.2019.00196`.

[4] G de Haan; A van Leest „Improved motion robustness of remote-PPG by using the blood volume pulse signature". In: 35 (2014), 9, pp. 1913–1926.

[5] Gerard de Haan; Vincent Jeanne „Robust Pulse Rate From Chrominance-Based rPPG". In: *IEEE Transactions on Biomedical Engineering* 60 (2013), 10, pp. 2878–2886.

[6] Guillaume Heusch; André Anjos; Sébastien Marcel „A Reproducible Study on Remote Heart Rate Measurement". In: (2017).

[7] Gee-Sern Hsu; ArulMurugan Ambikapathi; Ming-Shiang Chen. „Deep learning with time-frequency representation for pulse estimation from facial videos". In: *2017 IEEE International Joint Conference on Biometrics (IJCB)*. 2017, pp. 383–389. DOI: `10.1109/BTAS.2017.8272721`.

[8] Mayuko Kanemura. *What is RPPG?* `https://www.noldus.com/blog/what-is-rppg/`.

[9] Oleksii Kharkovyna. *Facial recognition: 8 Open-source tools to detect faces.* `https://medium.datadriveninvestor.com/facial-recognition-8-open-source-tools-to-detect-faces-4ec8e37bfcc6/`.

[10] Sungjun Kwon et al. „ROI analysis for remote photoplethysmography on facial video". In: *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2015, pp. 4938–4941. DOI: `10.1109/EMBC.2015.7319499`.

[11] Magdalena Lewandowska et al. „Measuring pulse rate with a webcam — A non-contact method for evaluating cardiac activity“. In: *2011 Federated Conference on Computer Science and Information Systems (FedCSIS)*. 2011, pp. 405–410.

[12] Xiaobai Li et al. „The OBF Database: A Large Face Video Database for Remote Physiological Signal Measurement and Atrial Fibrillation Detection“. In: *2018 13th IEEE International Conference on Automatic Face Gesture Recognition (FG 2018)*. 2018, pp. 242–249. DOI: 10.1109/FG.2018.00043.

[13] *MediaPipe Face Mesh*. https://google.github.io/mediapipe/solutions/face_mesh/.

[14] Ming-Zher Poh; Daniel McDuff; Rosalind Picard „Advancements in Noncontact, Multiparameter Physiological Measurements Using a Webcam“. In: *IEEE transactions on bio-medical engineering* 58 (2010). Pp. 7–11.

[15] Ronny Stricker; Steffen Müller; Horst-Michael Gross. „Non-contact video-based pulse rate measurement on a mobile service robot“. In: vol. 2014. Aug. 2014, pp. 1056–1062. DOI: 10.1109/ROMAN.2014.6926392.

[16] W. Wang; Sander Stuijk; Gerard Haan „A Novel Algorithm for Remote Photoplethysmography: Spatial Subspace Rotation“. In: *IEEE transactions on biomedical engineering* 0 (2015). P. 1.

[17] Wenjin Wang et al. „Algorithmic Principles of Remote PPG“. In: *IEEE Transactions on Biomedical Engineering* 64 (2017), 7, pp. 1479–1491.