Heart Rate Estimation Using rPPG

Salman and Chandra Prakash

Sardar Vallabhbhai National Institute of Technology, Surat, India
Computational Intelligence and Smart Motion Robotics (CISMR) Lab
salmanmalikcs026@gmail.com, cprakash@svnit.ac.in

Abstract. Remote Photoplethysmography (rPPG) is a non-invasive method that utilizes facial video recordings to estimate physiological parameters such as heart rate, offering an affordable and accessible solution for home healthcare. The system processes facial videos captured through standard webcams, extracting subtle changes in skin color using computer vision techniques like Gaussian pyramids to derive Photoplethysmography (PPG) signals. Heart rate is estimated by analyzing signals from facial regions of interest, incorporating multiple heart rate detection algorithms and advanced noise suppression techniques to improve accuracy and reliability. To validate the rPPG system, its results are compared with pulse oximeter measurements, yielding MAE and PCC values that fall within acceptable limits. With these enhancements, the rPPG system demonstrates promising reliability and robustness for real-time physiological monitoring.

Keywords: Remote Photoplethysmography · Heart Rate Estimation· Real Time Monitoring · Computer Vision.

1 Introduction

Photoplethysmography (PPG) measures blood volume changes with each heartbeat [4, 5].Extending this principle, Remote PPG (rPPG) enables non-contact monitoring using conventional RGB cameras—commonly integrated in smartphones and laptops—thus providing a more accessible alternative to pulse oximeters and cuff-based devices[7, 9]. From PPG signals, health parameters like heart rate (HR)and heart rate variability (HRV)) can be derived [6, 13].

On a global scale, cardiovascular diseases (CVDs) account for the highest mortality rates[8], regular monitoring of cardiovascular parameters using consumer-grade devices could enable early detection outside clinical environments [17, 16]. rPPG is also valuable in scenarios like COVID-19, where contactless home-based monitoring reduces infection risks. Obtaining reliable rPPG signals, however, remains challenging due to noise and environmental variability, prompting extensive research into improving signal quality.

Reflected light from facial regions such as the forehead and cheeks carries blood-flow information, which can be transformed into rPPG signals. There is, however, no agreement on an optimal extraction approach. Research contrasts

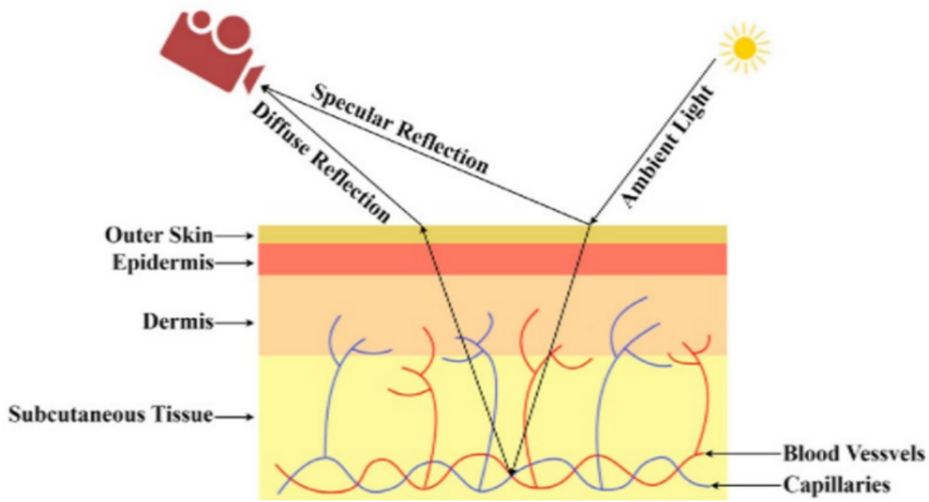

Fig. 1. Specular and diffuse reflection produced by the skin under environmental light.

methods including PCA, ICA, alternative color models, and normalization techniques, each showing varying levels of effectiveness depending on the dataset[3, 10].

Unlike contact-based PPG, which is robust due to direct skin contact, rPPG requires advanced signal processing to address motion and lighting artifacts. Recent methods employ complex algorithms to enhance accuracy, but often at the cost of computational efficiency. In this study, we propose a simple yet effective rPPG method using raw RGB video signals, avoiding heavy statistical or machine learning models, and benchmark it against established techniques.

2 Literature Survey

As a promising approach, Remote Photoplethysmography (rPPG) enables non-contact heart rate monitoring, unlike conventional techniques such as Electrocardiography (ECG) and wearable PPG that require direct physical contact. The literature presents numerous approaches for implementing rPPG, ranging from traditional signal processing methods to advanced deep learning techniques. This chapter reviews relevant work in the domain of non-contact heart rate measurement using rPPG, focusing on approaches that enhance accuracy, reliability, and real-time processing capabilities.

2.1 Video-Based Real-Time Monitoring

Mona Alnaggar et al. introduced a real-time video-based monitoring system that combines Eulerian Video Magnification (EVM) with the MediaPipe framework to estimate heart rate (HR) and respiration rate (RR) from facial videos [2]. Facial Regions of Interest (ROIs) are annotated using MediaPipe Face Mesh, followed by integral image analysis on RGB and HSV channels to estimate heart rate. The work emphasizes the critical role of ROI selection and color channel choice in improving the accuracy of rPPG signals derived from facial recordings.

2.2 Comprehensive Reviews on rPPG Methods

Hanguang Xiao et al. presented a comprehensive review of rPPG methods, categorizing them into traditional and deep learning-based approaches [18]. Research on rPPG has shifted from blind source separation (BSS) techniques, including Independent Component Analysis (ICA), to deep learning-based methods offering enhanced accuracy. Classical methods such as ICA and Principal Component Analysis (PCA) are capable of extracting heart rate signals from videos but face challenges under dynamic lighting and motion. Recent advances leverage deep learning approaches, particularly 2D CNNs and temporal models like GRUs, to overcome these issues and adapt effectively to real-world scenarios.

2.3 Early Contributions to rPPG

Lewandowska et al. developed an early system for measuring pulse rate using a webcam, employing Principal Component Analysis (PCA) for extracting the Photoplethysmographic signal from facial videos [12]. Their methodology utilized a Logitech Webcam to capture facial videos under natural light conditions, with ROI selection focusing on the forehead area. This approach was computationally efficient and demonstrated the feasibility of using PCA instead of ICA to reduce computational complexity without compromising signal quality. This study was one of the earliest to showcase the potential of webcams for pulse rate estimation, laying the groundwork for later developments in non-contact health monitoring.

2.4 RhythmEdge: Real-Time rPPG Estimation on Edge Devices

Zahid Hasan et al. introduced RhythmEdge, an edge computing system designed for real-time rPPG estimation using facial videos[12]. RhythmEdge utilizes a multi-task learning framework to estimate PPG signals and applies model compression techniques like pruning and quantization to optimize the system for resource-constrained edge devices. The system was benchmarked across multiple platforms, including NVIDIA Jetson Nano and Raspberry Pi, and achieved substantial reductions in latency and memory usage[14]. RhythmEdge is notable for its scalability and adaptability across different camera types and edge computing environments, making it suitable for large-scale deployment in telehealth applications [12].

2.5 Conventional Signal Processing Techniques

Traditional rPPG approaches utilize signal processing methods to extract heart rate from facial video data. In their work, Verkruysse et al. showed that consumer-grade cameras are effective for HR measurement, emphasizing the green channel because of its strong pulsatile response linked to hemoglobin absorption[15]. Subsequent techniques, such as POS and chrominance-based methods, sought to mitigate noise arising from motion artifacts and illumination changes. These methods typically employed a combination of bandpass filtering and spatial decomposition to enhance signal quality.

2.6 Deep Learning Based Approaches

Advancements in deep learning have enabled the creation of more robust and adaptive rPPG systems. Deep learning approaches, such as RhythmNet, incorporate convolutional layers to capture spatial information and GRU layers to model temporal relationships between frames, enhancing the robustness of the system under varying conditions of movement and illumination [14]. Another notable deep learning approach, DeepPhys, combines motion and appearance models to improve the system's ability to handle motion artifacts. These approaches significantly outperform traditional rPPG methods in terms of accuracy and adaptability[1].

2.7 Pyramid Methods-Based Approaches

The paper "Pyramid Methods in Image Processing" by Adelson[1]. presents a detailed ex- ploration of pyramid-based image processing techniques, emphasizing their multiresolution

capabilities. The authors introduce the image pyramid as a hierarchical data structure for effi- cient image analysis, compression, and enhancement. Pyramid representation facilitates spatial

localization and frequency decomposition, mirroring the multiscale processing in the human visual system. They discuss the Gaussian pyramid for low-pass filtering and the Laplacian pyramid for bandpass filtering, highlighting their ability to compress data, enhance features, and support efficient convolution operations. Applications range from pattern matching and texture analysis to creating seamless image mosaics and multifocus composites. The work underscores the versatility of pyramid structures in balancing computational efficiency with detailed spatial and frequency-based information processing, offering a robust framework for addressing diverse challenges in digital image processing[1].

3 Proposed Methodology

The approach encompasses video acquisition, facial landmark detection, region of interest (ROI) selection, signal extraction, preprocessing, frequency analysis, and performance evaluation. The methodology is designed to address challenges like motion artifacts, lighting variations, and facial occlusions, ensuring robustness in real-world settings. This expanded chapter includes detailed technical explanations, mathematical formulations, pseudocode algorithms, and additional subsections to provide a comprehensive understanding of the system.

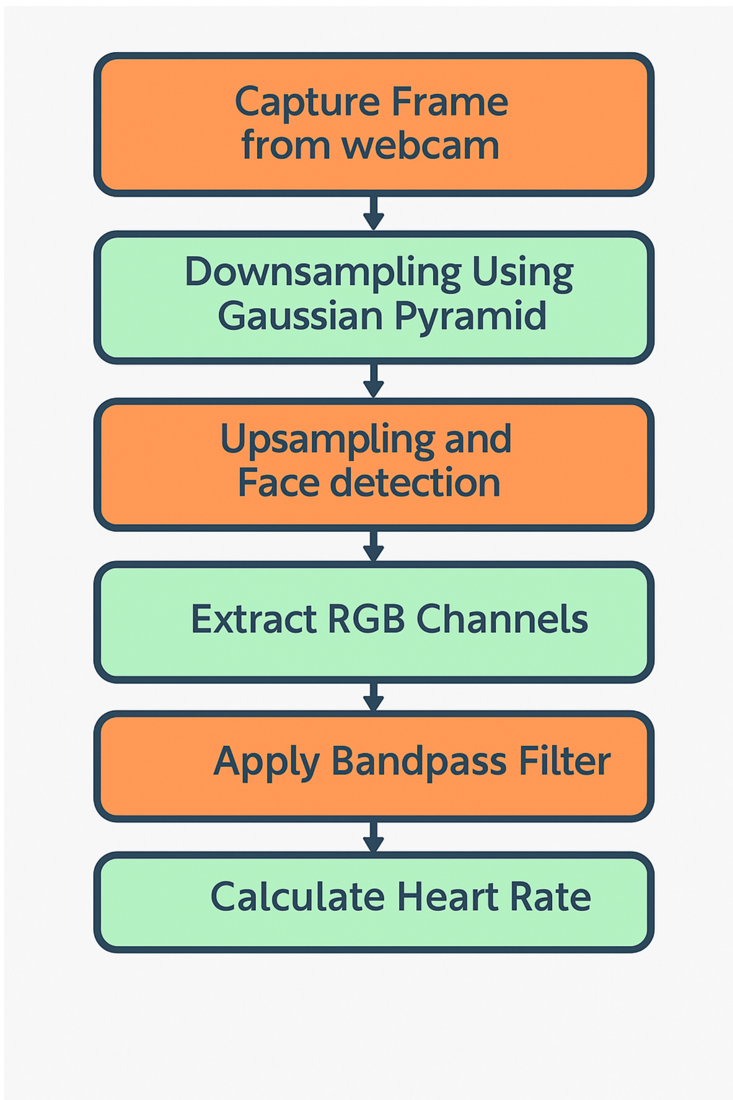

Fig. 2. Block diagram of proposed rPPG system.

3.1 Capture The Frames From Webcam

Webcam Setup and Video Capture: This component is responsible for setting up the video feed parameters, such as frame size, frame rate, and color channels. It ensures the system captures the necessary visual details while balancing computational efficiency. For this project, we used the built-in camera on the MacBook M1, which features a 720p FaceTime HD Camera. The camera supports a resolution of 1280x720 pixels and operates with a fixed frame rate of up to 30 frames per second (fps), providing sufficient quality for detecting subtle color changes required for rPPG. Additionally, the camera utilizes advanced image signal processing from the M1 chip, enhancing low-light performance and optimizing color balance, ensuring the video feed captures reliable data for accurate heart rate monitoring.
Face Detection and Analysis: The face detection module is a critical component of the system, responsible for identi- fying and locating human faces within the video feed. This step narrows down the area for analysis, ensuring that processing resources are focused only on regions with the most relevant information, such as the facial area, while ignoring irrelevant background data. Accurate face detection is essential for enabling precise extraction of heart rate signals, as it isolates key regions like the forehead, which provide reliable photoplethysmographic signals. To implement this module, we utilized Python libraries such as cv2 (OpenCV), cvzone, and a dedicated face detection module.

3.2 Downsampling Using Gaussian Pyramid

Gaussian Pyramid : The Gaussian Pyramid is a multi-scale hierarchical representation used in image processing, where the image is repeatedly down-sampled to generate progressively smaller and smoother versions. This approach facilitates efficient analysis of features at various resolutions and is commonly applied in tasks such as image compression, object detection, and computer vision applications, including remote photoplethysmography(rPPG)[1].
Gaussian Pyramid Concept and Construction The Gaussian Pyramid is created by iteratively applying Gaussian smoothing to an image followed by down-sampling, producing a sequence of images that progressively decrease in size and smoothness.
1 - Gaussian Smoothing : A Gaussian filter is applied to the image to reduce high-frequency noise and detail, resulting in a blurred version of the image.
2 - Down-Sampling : The blurred image is subsampled by removing rows and columns, effectively halving the width and height of the image.
3 - Repetition : Steps 1 and 2 are repeated multiple times to generate successive levels of the pyramid, each representing the image at a coarser scale.

Mathematics in Gausian Pyramid Method :

The Gaussian Pyramid can be expressed mathematically as:

$Gi+1\ (x, y) = \mathrm{DownSample}(\mathrm{GaussianFilter}(Gi(x, y)))$

Where :

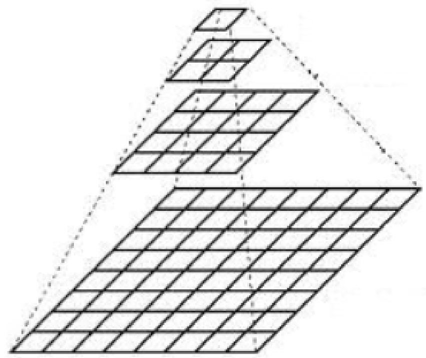

Fig. 3. Visualization of the Gaussian Pyramid.

- Gi(x, y): The image at level i in the pyramid.
- GaussianFilter: A 2D Gaussian function applied to the image.
- DownSample: The process of reducing the image dimensions by a factor of 2.
  The Down Sampling can be mathematically represented as:

$$g_l(i,j) = \sum_{m=-2}^{2} \sum_{n=-2}^{2} w(m,n)\, g_{l-1}(2i+m,\, 2j+n)$$

Explanation of Notation :

- $g_l(i,j)$: The pixel value at coordinates (i,j) in the image at level l (current pyramid level).
- g_{l-1}: The image at level $l-1$ (the previous, higher-resolution level in the pyramid).
- $w(m,n)$: The weights of the Gaussian filter, which define how neighboring pixels contribute to the smoothing operation. These weights are determined by the Gaussian function.
- m,n: The offsets that define a **5x5** neighborhood around the current pixel in the original image.

3.3 Upsampling

Upsampling is the process of increasing the resolution of an image, typically after it has been downsampled. It restores the image to its original size or a higher resolution. In the context of Gaussian pyramids, upsampling is the reverse operation of downsampling and is used to reconstruct or visualize data at a higher resolution. //

Mathematical Formula for Upsampling :

$$g_{l,n}(i,j) = \sum_{p=-2}^{2} \sum_{q=-2}^{2} w(p,q)\, g_{l-1,n}\left(\frac{i-p}{2}, \frac{j-q}{2}\right) \tag{1}$$

Explanation of Notation :

- $g_{l,n}(i,j)$: The pixel value at coordinates (i,j) in the higher-resolution image at level l.
- $g_{l-1,n}$: The lower-resolution image at the previous level $(l-1)$ of the Gaussian Pyramid.
- $w(p,q)$: The weights of the Gaussian filter, which determine how neighboring pixels contribute to the value at $g_{l,n}(i,j)$. These weights are derived from a 2D Gaussian kernel.
- $\frac{i-p}{2}, \frac{j-q}{2}$: The pixel coordinates in the lower-resolution image $(g_{l-1,n})$. This maps the higher-resolution coordinates (i,j) to their corresponding positions in $g_{l-1,n}$ after accounting for upsampling.

3.4 Extract Region of Interest (ROI)

Selection of Optimal ROIs

Forehead: The forehead is free from motion artifacts caused by facial expressions or speech. It provides a relatively flat surface, making it less prone to shadows or occlusions. It is rich in capillaries, leading to pronounced blood volume changes visible as subtle color variations.

Cheeks: The cheeks are highly vascularized and exhibit significant blood volume changes. They provide a larger area for averaging pixel values, which improves signal-to-noise ratio (SNR). Cheeks are less likely to be occluded by accessories such as glasses, compared to other facial regions.

Steps for ROI Extraction

Face Detection Face detection algorithms, such as Haar Cascade, DNN-based detectors, or MediaPipe, are employed to identify the face within a video frame, producing a bounding box that encompasses the entire face.

Landmark Detection Facial landmarks are detected to precisely locate regions such as the forehead, eyes, nose, and cheeks. Algorithms such as Dlib's facial landmark detector or MediaPipe Face Mesh are commonly used for this purpose. Landmarks provide key points like the forehead region, which lies above the eyes and between the eyebrows, and the cheek regions, which lie below the eyes and lateral to the nose.

ROI Definition Using the detected landmarks, rectangular or polygonal ROIs are defined for the forehead and cheeks. The forehead ROI is a rectangle or trapezoid extracted above the eyebrows, spanning the middle of the forehead. The cheek ROIs consist of two rectangular regions extracted below each eye and to the sides of the nose.

Signal Extraction from ROIs For each ROI, the average pixel intensity is computed across the RGB channels for every frame. These temporal signals form the basis for further processing. The signals from the forehead and both cheeks are either analyzed separately or combined to improve robustness.

3.5 Extract "RGB" Channels

The RGB (Red, Green, Blue) channels play a pivotal role in remote Photoplethysmography (rPPG) for extracting physiological signals, particularly heart rate. These color channels are derived from the captured facial video and contain crucial information about subtle color variations caused by blood flow beneath the skin.

Importance of RGB Channels in rPPG

Physiological Basis Blood absorbs light differently depending on its oxygenation level. This causes minor variations in skin color that are often imperceptible to the naked eye. The Red and Green channel is especially sensitive to hemoglobin absorption, making it the most informative for detecting changes in blood volume [12]. The Blue channel, while less dominant, provides complementary information that can improve robustness and noise suppression when combined with the Green channel [12].

Signal Composition The temporal changes in the intensity of the RGB channels correlate with the cardiac cycle, allowing the extraction of periodic signals that represent the heart rate.

Noise Resilience By analyzing all three channels, rPPG systems can mitigate noise caused by lighting conditions, movement artifacts, and skin tone variations, thereby improving the accuracy and robustness of the heart rate estimation.

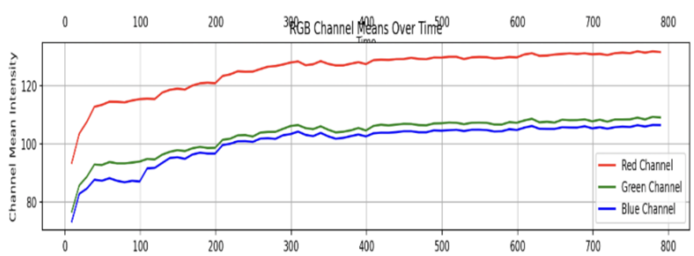

Fig. 4. Diagram showing the RGB Channels

3.6 Apply Fourier Transformation

The Fourier Transform is a mathematical method that converts a time-domain signal into its frequency-domain representation. For heart rate signal analysis, this transformation is essential for detecting the periodic components corresponding to the cardiac rhythm, decomposing the signal into sinusoidal elements, each with a specific frequency and amplitude.

3.7 Purpose of Fourier Transform in Heart Rate Analysis

1. **Purpose of Fourier Transform in Heart Rate Analysis:**
   - Heart rate signals often contain noise and irrelevant information, making it difficult to interpret them directly from the time domain. By transforming the signal into the frequency domain, the dominant frequencies corresponding to the heartbeats can be easily identified.
   - It highlights periodic patterns that are otherwise obscured in the time domain. For instance, the heart rate, which is typically between 60 to 120 beats per minute, corresponds to frequencies between 1 and 2 Hz.
2. **Steps to Perform Fourier Transform:**
   - **Signal Preprocessing:** Before applying the Fourier Transform, the raw signal is usually detrended and smoothed to eliminate baseline wander and abrupt spikes.
   - **Discrete Fourier Transform (DFT):** The actual transformation can be performed using algorithms like the Fast Fourier Transform (FFT), which efficiently computes the frequency spectrum of the signal.
   - **Interpretation of Results:** The transformed data is visualized as a frequency spectrum. Peaks in this spectrum represent dominant frequencies in the signal, which correspond to the heart rate and other physiological phenomena.

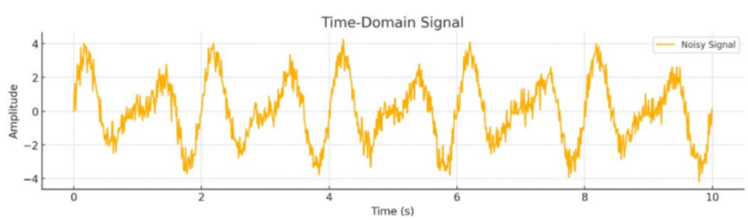

Fig. 5. Time-Domain Signal: A noisy periodic signal plotted as amplitude vs. time. This represents how the raw signal (e.g., rPPG or PPG) fluctuates with physiological changes and external noise.

3.8 Applying Bandpass Filter

Bandpass filtering [11] is a crucial step in heart rate monitoring, used to isolate the frequency range corresponding to cardiac activity while suppressing noise.

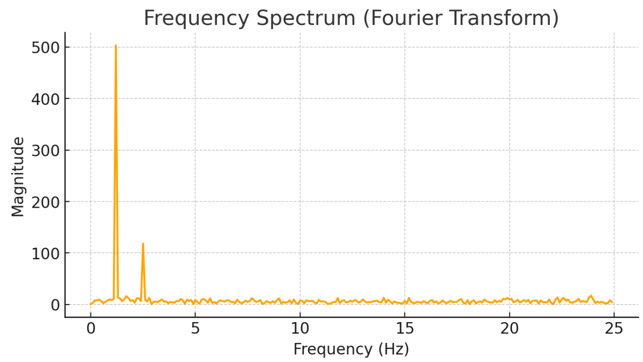

Fig. 6. Frequency Domain Signal obtained using Fourier Transform. The two peaks represent the fundamental heartbeat frequency and its harmonic.

Since typical adult heart rate signals fall between 1–2 Hz (60–120 beats per minute), the filter passes only this range, removing low-frequency components such as baseline drift caused by respiration or motion, as well as high-frequency noise from electrical interference or sensor artifacts.

The procedure begins with acquiring the raw signal, often from a photoplethysmogram (PPG) sensor. The Fast Fourier Transform (FFT) is then applied to convert the signal into the frequency domain, where peaks within the 1–2 Hz range indicate the heart rate. A bandpass filter, commonly designed using Butterworth or Chebyshev techniques, is applied to isolate these frequencies. In real-time applications, IIR or FIR filters are implemented for efficient processing, often using tools like MATLAB, Python (SciPy), or specialized DSP hardware.

Once filtering is complete, the signal is transformed back to the time domain, producing a cleaner waveform that highlights the heartbeat components. Combining Fourier analysis with bandpass filtering ensures the signal retains only the most relevant frequency components, enabling accurate and reliable detection of heart rate and further analysis such as heart rate variability (HRV).

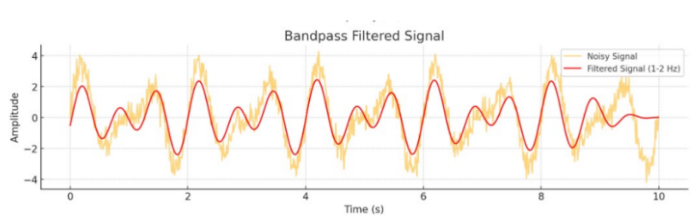

Fig. 7. Band-pass Filtered Signal

3.9 Calculate Heart Rate :

Peak Detection
To determine heart rate, the algorithm detects peaks in the filtered signal, which correspond to periodic variations in blood volume. The procedure involves the following steps:

- **Find Local Maxima:** A point s_i is considered a peak if:

$$s_i > s_{i-1} \quad and \quad s_i > s_{i+1}$$

  This can be implemented programmatically using libraries such as `scipy.signal.find_peaks`.
- **Filter Peaks by Amplitude:** A threshold is applied to exclude smaller peaks caused by noise or motion artifacts, ensuring only physiologically relevant peaks are considered.

Heart Rate Calculation
Once peaks are identified, the heart rate is calculated by analyzing the intervals between consecutive peaks:

- **Time Intervals Between Peaks:** Let t_i represent the time of the i^{th} peak. The intervals between peaks are:

$$\Delta t_i = t_{i+1} - t_i$$

- **Heart Rate Formula:** The heart rate in beats per minute (BPM) is given by:

$$HeartRate = \frac{60}{Mean(\Delta t)}$$

  Alternatively, using the dominant frequency f_{peak} from the signal's frequency spectrum:

$$HeartRate = f_{peak} \times 60$$

This approach provides an accurate estimation of heart rate based on the periodicity of the filtered signal.

3.10 Results

The system was tested in real time using a webcam, and estimated heart rates were compared with a pulse oximeter. The Mean Absolute Error (MAE) was within acceptable limits, confirming the reliability of the proposed rPPG method.

Mean Absolute Error (MAE) The MAE is given by:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{2}$$

where y_i is the actual value, $\hat{y}_i$ the predicted value, and n the number of samples.

Actual HR	rPPG	Actual HR	rPPG	Actual HR	rPPG
97	101.2	100	101.0	86	96.3
96	103.6	100	102.3	88	99.2
97	100.2	99	103.7	88	93.2
88	86.2	98	104.2	87	94.3
89	87.6	99	104.1	85	88.9
88	82.1	98	105.7	86	91.3
87	81.2	99	110.0	86	85.4
89	82.5	99	110.7	87	83.8
87	85.7	98	109.8	86	90.0
86	89.9	87	94.2	84	87.9
101	94.1	87	93.6	83	88.1
101	97.0	86	96.3	84	85.6
100	101.1	88	99.2	76	73.4
100	101.5	88	93.2	76	75.1
101	101.0	87	94.3	76	80.1
83	83.8	85	88.9	77	81.1
77	76.1	86	91.3	78	80.9
77	74.7	86	85.4		

Table 1. Heart rate results comparison between pulse oximeter (Actual HR) and rPPG model (Calculated Results).

Pearson Correlation Coefficient (PCC) The PCC is defined as:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2} \quad (3)$$

where x_i and y_i are the sample points, $\bar{x}$ and $\bar{y}$ are their means, and $r \in [-1, 1]$.

Table 2. Summary of Mean Absolute Error and Pearson Correlation Coefficient

Mean Absolute Error	Pearson Correlation Coefficient
4.745	0.877

4 Conclusion

The proposed rPPG system successfully detects heart rate from facial videos. The use of PCA for dimensionality reduction made the system computationally efficient, allowing for real-time processing. Results indicate that rPPG can be a viable method for non-contact heart rate monitoring, especially in home care environments.

5 Future Work

Future work will aim to enhance the system's robustness to varying lighting conditions and subject motion, as well as to enable accurate heart rate detection from multiple individuals simultaneously, thereby improving its practical applicability in real-world scenarios

References

1. Adelson, E.H., Anderson, C.H., Bergen, J.R., Burt, P.J., Ogden, J.M.: Pyramid methods in image processing. RCA engineer **29**(6), 33–41 (1984)
2. Alnaggar, M., Siam, A.I., Handosa, M., Medhat, T., Rashad, M.: Video-based real-time monitoring for heart rate and respiration rate. Expert Systems with Applications **225**, 120135 (2023)
3. Boccignone, G., Conte, D., Cuculo, V., d'Amelio, A., Grossi, G., Lanzarotti, R.: An open framework for remote-ppg methods and their assessment. Ieee Access **8**, 216083–216103 (2020)
4. Elgendi, M.: On the analysis of fingertip photoplethysmogram signals. Current cardiology reviews **8**(1), 14–25 (2012)
5. Elgendi, M.: PPG signal analysis: An introduction using MATLAB®. CRC press (2020)
6. Elgendi, M., Fletcher, R., Liang, Y., Howard, N., Lovell, N.H., Abbott, D., Lim, K., Ward, R.: The use of photoplethysmography for assessing hypertension. NPJ digital medicine **2**(1), 60 (2019)
7. Frey, L., Menon, C., Elgendi, M.: Blood pressure measurement using only a smartphone. NPJ digital medicine **5**(1), 86 (2022)
8. Group, A.W., of Cardiology, E.S., et al.: European society of cardiology: cardiovascular disease statistics 2021. European heart journal **43**(8), 716–799 (2022)
9. Haugg, F., Elgendi, M., Menon, C.: Assessment of blood pressure using only a smartphone and machine learning techniques: A systematic review. Frontiers in Cardiovascular Medicine **9**, 894224 (2022)
10. Haugg, F., Elgendi, M., Menon, C.: Effectiveness of remote ppg construction methods: a preliminary analysis. Bioengineering **9**(10), 485 (2022)
11. He, L., Alam, K.S., Ma, J., Burkholder, E., Chung Chu, W.C., Iqbal, A., Ahamed, S.I.: Remote photoplethysmography heart rate variability detection using signal to noise ratio bandpass filtering. In: 2021 IEEE International Conference on Digital Health (ICDH). pp. 133–141 (2021). https://doi.org/10.1109/ICDH52753.2021.00025
12. Lewandowska, M., Rumiński, J., Kocejko, T., Nowak, J.: 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). pp. 405–410. IEEE (2011)
13. Lin, W.H., Wu, D., Li, C., Zhang, H., Zhang, Y.T.: Comparison of heart rate variability from ppg with that from ecg. In: The International Conference on Health Informatics: ICHI 2013, Vilamoura, Portugal on 7-9 November, 2013. pp. 213–215. Springer (2014)
14. Nadrag, C., Poenaru, V., Suciu, G.: Heart rate measurement using face detection in video. In: 2018 international conference on communications (COMM). pp. 131–134. IEEE (2018)

15. Poenaru, C., Suciu, G.: Heart rate measurement using face detection in video. In: Conference Paper (2018)
16. Rocque, G.B., Rosenberg, A.R.: Improving outcomes demands patient-centred interventions and equitable delivery. Nature Reviews Clinical Oncology **19**(9), 569–570 (2022)
17. Subbiah, V.: The next generation of evidence-based medicine. Nature medicine **29**(1), 49–58 (2023)
18. Xiao, H., Liu, T., Sun, Y., Li, Y., Zhao, S., Avolio, A.: Remote photoplethysmography for heart rate measurement: A review. Biomedical Signal Processing and Control **88**, 105608 (2024)