Robust PPG Peak Detection Using Dilated Convolutional Neural Networks

Kianoosh Kazemi [1,1,1], Juho Laitala [2], Iman Azimi [2], Pasi Liljeberg [2], and Amir M. Rahmani [2]

[1]University of Turku
[2]Affiliation not available

November 8, 2023

Abstract

Accurate peak determination from noise-corrupted photoplethysmogram (PPG) signal is the basis for further analysis of physiological quantities such as heart rate and heart rate variability. In the past decades, many methods have been proposed to provide reliable peak detection. These peak detection methods include rule-based algorithms, adaptive thresholds, and signal processing techniques. However, they are designed for noise-free PPG signals and are insufficient for PPG signals with low signal-to-noise ratio (SNR). This paper focuses on enhancing PPG noise-resiliency and proposes a robust peak detection algorithm for noise and motion artifact corrupted PPG signals. Our algorithm is based on Convolutional Neural Networks (CNN) with dilated convolutions. Using dilated convolutions provides a large receptive field, making our CNN model robust at time series processing. In this study, we use a dataset collected from wearable devices in health monitoring under free-living conditions. In addition, a data generator is developed for producing noisy PPG data used for training the network. The method performance is compared against other state-of-the-art methods and tested in SNRs ranging from 0 to 45 dB. Our method obtains better accuracy in all the SNRs, compared with the existing adaptive threshold and transform-based methods. The proposed method shows an overall precision, recall, and F1-score 80%, 80%, and 80% in all the SNR ranges. However, these figures for the other methods are below 78%, 77%, and 77%, respectively. The proposed method proves to be accurate for detecting PPG peaks even in the presence of noise.

Posted on 10 Dec 2020 — CC-BY 4.0 — https://doi.org/10.36227/techrxiv.16529310.v3 — This is a preprint. Version of Record available at https://doi.org/10.3390/s22166054

Robust PPG Peak Detection Using Dilated Convolutional Neural Networks

Kianoosh Kazemi, Juho Laitala, Iman Azimi, Pasi Liljeberg, Amir M. Rahmani

Abstract

Accurate peak determination from noise-corrupted photoplethysmogram (PPG) signal is the basis for further analysis of physiological quantities such as heart rate and heart rate variability. In the past decades, many methods have been proposed to provide reliable peak detection. These peak detection methods include rule-based algorithms, adaptive thresholds, and machine learning techniques. However, they are designed for noise-free PPG signals and are insufficient for PPG signals with low signal-to-noise ratio (SNR). This paper focuses on enhancing PPG noise-resiliency and proposes a robust peak detection algorithm for noise and motion artifact corrupted PPG signals. Our algorithm is based on Convolutional Neural Networks (CNN) with dilated convolutions. Using dilated convolutions provides a large receptive field, making our CNN model robust at time series processing. In this study, we use a dataset collected from wearable devices in health monitoring under free-living conditions. In addition, a data generator is developed for producing noisy PPG data used for training the network. The method performance is compared against other state-of-the-art methods and tested in SNRs ranging from 0 to 45 dB. Our method obtains better accuracy in all the SNRs, compared with the existing adaptive threshold and transform-based methods. The proposed method shows an overall precision, recall, and F1-score 80%, 80%, and 80% in all the SNR ranges. However, these figures for the other methods are below 78%, 77%, and 77%, respectively. The proposed method proves to be accurate for detecting PPG peaks even in the presence of noise.

Photoplethysmogram Peak detection Convolutional Neural Network Wearable devices Motion artifacts

1 Introduction

There is a growing demand for ubiquitous health monitoring systems. These systems are developed to provide proactive healthcare solutions as well as reduce medical costs: e.g., providing efficiency and cost-savings for doctors, nurses, and pharmaceutical companies [1]. Fortunately, rapid advancement in the Internet of Things (IoT)-based systems and wearable devices offer opportunities for the development of health monitoring systems [2]. Such IoT-based healthcare systems can provide comprehensive patient care by leveraging various sensor types, communication units, and computing resources. Wearable electronics - such as wristbands and smart rings - enable the ubiquitous collection of biomedical signals, including electrocardiogram (ECG) and photoplethysmogram (PPG) [3].

PPG is a low-cost, non-invasive, and simple optical technique used for measuring the synchronous blood volume changes in tissue such as the surface of the finger, toe, wrist, and forehead [3]. This approach is widely used in wearable IoT-based applications due to its high level of feasibility and ease of measurement [4]. Collected PPG signals can be used to extract various health parameters, such as heart rate and heart rate variability. These health parameters are obtained by the determination of the systolic peaks in the PPG records. However, the quality of the PPG waveform is easily affected by surrounding noises such as background noises and motion artifacts. As users engage in a variety of physical activities, these noises are unavoidable in IoT-based healthcare systems. Subsequently, when the signal quality is poor (i.e., low signal-to-noise ratio (SNR)), accurate detection of peaks in PPG signals becomes challenging. This issue increases false peak determination, which results in inaccurate vital signs extraction.

Numerous studies have been proposed to determine the PPG signal peaks accurately. In some specified cases, the signal is inspected manually by experts, and then the location of the peaks is annotated. These methods are often implemented in hospitals and clinics and are mainly used as gold standard methods for validation [5]. However, implementation of such methods requires much time and domain knowledge, which is not feasible.

On the other hand, there are various automatic techniques employing different signal processing, filtering, and rule-based techniques. These methods mainly include adaptive threshold [6], transform-based techniques [7], derivative calculation [8], and computer-based filtering [9].

Adaptive threshold techniques are commonly used for peak detection. In these methods, a threshold is set based on the proceeding peaks amplitude, and the threshold constantly decays until it reaches the next peak. Then, the value of the threshold is updated, and the threshold is controlled by different features such as duration, amplitude, beat-to-beat- intervals, and sampling frequency [10], [11]. Transform-based techniques are other alternatives for PPG peak detection. These methods mostly employ signal processing algorithms such as discrete wavelet transform [12], stationary wavelet transform [7] and Hilbert transform [13]. Wavelet-based techniques are mostly used in preprocessing stage for denoising. These techniques decompose the signal into multiple subbands which have the same resolution as the original signal. Then, by composing the desire subbands, the informative part of the signal is regenerated, and the baseline wanders and high-frequency (HF) noises are eliminated. In contrast, Hilbert transform is utilized for envelope extraction and peak detection task. Hilbert transform is a powerful tool in analyzing the amplitude and frequency of a signal instantaneously. [13], and [14] indicate that the zero-crossing points in the Hilbert transform correspond to the peak locations. However, these methods are insufficient for noise-contaminated signals, and they become unreliable if the SNR drops below a certain level.

In addition, machine learning-based approaches have been developed for PPG signal analysis [15], [16]. For example, a three-layered feed-forward neural network was introduced in [17] for PPG peak detection. The method was only trained and evaluated with low-noise signals.

The conventional peak detection techniques in the literature are mainly designed for noise-free or low-noise PPG signals. Therefore, they are insufficient to determine PPG peaks' locations when the signal quality is poor due to motion artifacts and HF noises. These noises are inevitable in wearable-based health and well-being monitoring systems. We believe that a peak detection method is required to determine systolic peaks in noisy PPG, leveraging temporal information in the signal. The robustness of such a method requires to be investigated against different noise levels.

In this paper, we propose a CNN-based peak determination approach for PPG signals with different levels of motion artifacts. The convolution layers in our network are dilated, resulting in a high receptive field efficiency. Therefore, the network can use temporal information in PPG peak detection and learn complex problems associated with the noisy PPG signals. Our analysis exploits PPG signals and motion artifacts collected by wearable devices in health monitoring under free-living conditions. We develop a generator function to produce PPG signals with a wide range of noise, augmenting the training data and creating noisy signals similar to real-life PPG records. Using the PPG signals, the proposed method is evaluated in comparison with state-of-the-art PPG peak detection methods. In summary, the major contributions of the paper are as follows:

- Proposing a dilated convolutional neural networks for addressing the problem of PPG peak detection in the presence of noise.
- Developing a generator function for producing PPG signals with different noise levels to be used for model training.
- Assessing the robustness of the proposed method using noisy PPG signals with SNRs ranging from 0 to 45 dB.
- Evaluating the proposed method in terms of accuracy compared to conventional methods, including adaptive threshold and Hilbert transform.
- Providing the model implemented in Python for the community to be used in their solutions [1].

The rest of the paper is organized as follows. The background and related work of this research is outlined in Section 2 and 3. We introduce the dataset used in this work in Section 4. Section 5 describes the development of the proposed method in detail. In Section 6, we evaluate our method in comparison with other published methods. Finally, Section 7 concludes the paper.

[1] `https://github.com/HealthSciTech/Robust_PPG_PD`

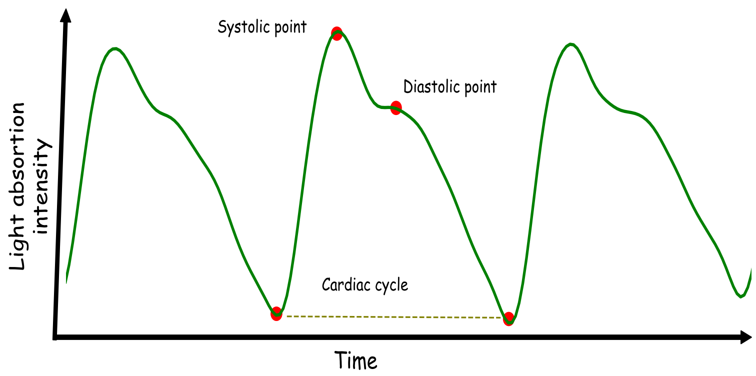

Figure 1: An example of a filtered PPG signal.

2 Background

In this section, we briefly describe PPG and neural networks proposed for PPG-based applications.

2.1 Photoplethysmography

PPG is a convenient method for sensing the blood flow rate at peripheral sites. Therefore, this signal can be used to determine the cardiac cycle [3]. The PPG sensor includes two main components: i.e., a light source and a photodetector. PPG signals are acquired by emitting light in different wavelengths (e.g., infrared, red, and green, often at 940, 660, and 550 nanometers, respectively) to the skin surface and capturing the reflected light via photodetectors. The infrared and red lights are commonly used for measuring heart rate and blood oxygen saturation. Furthermore, the green light is widely used in wearable devices such as smartwatches [18].

The variation in the PPG signal is associated with cardiac and respiration oscillations. Figure 1 indicates a view of a PPG signal, where the heart rate values can be estimated by measuring the difference of the time interval between two successive peaks. The signal consists of two main components: i.e., the alternating current (AC) and direct current (DC). The AC part denotes synchronous cardiovascular fluctuations caused by cardiac activity, while the DC portion denotes various low-frequency elements of the blood flow, such as respiration [3], [16].

2.2 Neural Networks in PPG Applications

Artificial neural networks are inspired by the human brain and imitate how biological neurons interact with one another, comprising an input layer, hidden layers, and an output layer [19]. Neural networks algorithms have been recently used in various PPG signal applications. In [20], a classification method based on multilayer perceptron (MLP) network was presented. In this study, an MLP network was trained to classify the pattern of the onset and systolic of the PPG signals with different window sizes. The preprocessing stage includes two steps: i.e., signals segmentation and smoothing using a simple mean square regression. Then, the results are fed to the network as features for pattern recognition. Chen *et al.* [21] proposed a hidden Markov model for PPG classification. They first used linear predictive coding and sample entropy methods to extract different features from the PPG waveforms. Then, a vector quantization method was employed to convert the features into the prototype vectors which were utilized to estimate the parameters for hidden Markov model parameters. Reiss *et al.* [22] introduced a CNN architecture for heart rate estimation. In this study, the PPG signals and corresponding three-axis accelerometer data were used to train the model.

For PPG noise removal, Xu *et al.* [15] proposed a deep recurrent neural network and stochastic modeling recover the noise-corrupted PPG signals. They first used recurrent neural networks for segmentation. Then, a Kalman filter was employed to extract clean PPG and create a stochastic model. They also tested their method on a real-time dataset acquired by a wearable glove. In addition to noise removal, deep learning methods were proposed for PPG quality assessment [16, 23]. In these studies, 1D and 2D CNN models were trained to discriminate between reliable and unreliable signals. The methods were evaluated by comparing the result with ECG references.

3 Related Work

In this section, we describe several PPG peak detection methods with different complexities which have been developed in the last decades. Most peak detection methods contain two main stages: i.e., 3.1. preprocessing (or filtering) and 3.2. envelope detection, and peak determination.

3.1 Preprocessing

Preprocessing is one of the significant stages in the PPG peak detection task. This step aims to remove the frequency component of the signal that does not reflect the fundamental features. In the preprocessing step, different filtering methods –such as low-pass filter, high-pass filter, singular value decomposition, and mode decomposition– are employed to suppress the baseline distortion and HF noises. Such methods enlarge the systolic peak part of the PPG signals. Tran *et al.* [24] proposed low-pass and high-pass filtering methods with cut-off frequencies of 0.4 and 8 Hz to remove the motion artifact and HF noise, respectively. In another study, [25], a low-pass filter with a cut-off frequency of 15 Hz was used for noise cancellation. A high-pass filter with a cut-off frequency of 0.01 Hz was also chosen to suppress the baseline wandering. Ricardo Ferro [13] used a digital fourth-order Chebyshev band-pass filter with the bandwidth of 0.5–16 Hz for eliminating the DC and HF components in the PPG signal. The major component of background noise is presented in the frequency range of 0.15 to 5.0 Hz. This was achieved by employing a band-pass filtering method with cut-off frequencies of 0.5 and 5.5 Hz [26]. Prieto *et al.* [27] used a combination of two zero-phase delay fourth-order high-pass and eighth-order low-pass Butterworth filters with a bandwidth of $0.1 - 16$ Hz to remove unwanted signals. Moving average filters were also utilized in noise surpassing. For instance, in [8], a 3-point bidirectional moving average was proposed to remove the phase delay caused by the filter.

In [28], a variational mode decomposition was used to enhance the signal quality and suppress the motion artifact. The decomposition was implemented in two stages to minimize the balancing error. Another method for noise removal is empirical mode decomposition which is computationally expensive. In this method, the signal is adaptively decomposed into intrinsic mode functions, and then by using averaging, the noise components are eliminated [9]. Paradkar *et al.* [7] introduced a singular value decomposition along with a moving average filter to extract the periodic component from the raw signal and reduce background noise.

3.2 Envelope extraction and peak determination

This stage generally includes extracting different features such as the maxima and minima, slope of the signal, and signals' envelope using well-known and robust algorithms, e.g., adaptive threshold, transform-based, and machine learning-based algorithms to determine the signal peaks.

3.2.1 Adaptive threshold

The adaptive threshold is a common technique used for PPG peak detection. This method employs a constant specified by signals' temporal and frequency domain features and time intervals. The constant could be decaying or growing due to the dynamic nature of the PPG waveform [6], [24]. For example, Shin *et al.* [29] proposed to update the threshold according to different features such as the sampling frequency, preceding peaks, and standard deviation of the signal. In [10], the adaptive thresholding is equipped with a morphological filter to remove the low noises and use a slope sum function to pinpoint the location of peaks accurately. Van Gent [30], [31] presented a method based on an adaptive threshold followed by moving average and spline interpolation methods if the detected peaks show clipping. The complexity of the adaptive threshold PPG peak detection methods is low. However, the methods are sensitive to noise and fail to accurately identify the peaks when the PPG signal is contaminated by noise. In other words, the PPG signal changes rapidly due to noise, so the methods are incapable of selecting appropriate thresholds.

3.2.2 Transform-based techniques

In addition, transform-based techniques are developed for PPG peak detection. These methods are mainly based on various non-linear transformations such as wavelet [7] and Hilbert [13] transforms, by which the signal's temporal and frequency domain features are extracted. Then, various thresholds –such as zero-crossing points or decision logic– are set to extract the signals component and corresponding peaks in the original signals. In [13], a Hilbert transform was accompanied by moving average and Shannon energy envelope techniques to locate the position of the systolic peaks. Vadrevu *et al.* [32] introduced a stationary wavelet transform to extract two sets of coefficients from the PPG signal. Then, using multiscale sum and product, the peaks' sharpness was enhanced in the edges, and the other values remained near zero. Following that, the zero-crossing points were extracted to obtain the locations of the systolic peaks. Leveraging transform based-methods, the peak positions can be detected more accurately. For instance, Chakraborty *et al.* [33] proposed a robust algorithm –enabled by a Hilbert transform, amplitude thresholding, and signal derivative– to detect PPG systolic peaks. Their algorithm achieved a better performance in comparison to an adaptive threshold technique. However, the transform-based methods are still insufficient for wearable-based PPG, as they fail to precisely determine systolic peaks' in distorted PPG signals.

In another work, Jang *et al.* [11] introduced a positioning algorithm to locate PPG peaks. The method includes denoising and abnormal intervals removal steps. In [34], PPG peaks were automatically detected and corrected, exploiting a Poincare plot feature and envelope detection. The methods mentioned above are accurate when the PPG signal quality is good. However, they are highly susceptible to motion artifacts and environmental noises. They fail to differentiate false noisy peaks and systolic peaks and subsequently result in inaccurate peak detection. Moreover, the probability of false peak detection increase in signals with high heart rate. Consequently, these methods are insufficient for wearable-based monitoring, in which the users might engage in various physical activities.

3.2.3 Machine learning methods

Traditional machine learning and deep learning methods have been employed to analyze cyclostationary biosignals such as PPG and ECG. For example, Xiang *et al.* [35] proposed 1-D CNN for QRS complex detection. In the preprocessing stage, a derivative function followed by an averaging system was used for noise removal. Then, the signals were fed to the CNN method for automatic feature extraction and classification. In [36], a faster R CNN model was proposed for ECG peak detection. Their method included three steps. First, the ECG signals were segmented and transformed into 2D images. Second, the images were fed to the model, and the output features map was put into a regional proposal network. In the final step, by setting a threshold, low probability outputs were excluded. Their method was tested with 24-hours wearable ECG recordings. In addition, Laitala *et al.* [37] proposed an automatic R-peak detection for ECG signals. The method comprised a bidirectional LSTM to obtain the probabilities and locations of R-peaks. The machine-learning-based methods mentioned above have been utilized for ECG R-peak detection. They are insufficient for PPG signals due to the difference in the signals' origins. PPG systolic peaks detection is more challenging as the peaks' slopes are not as large as QRS-complex in ECG. PPG signal quality and the waveform are also highly susceptible to artifacts generated, for example, by the user's hand movements.

For PPG peak detection, Sumukha *et al.* [17] proposed an online sequential learning algorithm. Their method included two steps. First, they divided the PPG signals into a set of fundamental sinusoidal defined segments. Among these segments, only one segment contains a peak. In the second step, a feedforward neural network method was trained to detect peaks in the segments. However, the method could not differentiate systolic peaks with noise peaks, so it might fail with noisy signals. Moreover, the evaluation was merely limited to noise-free and low-noise PPG signals.

Table 1: Background information of the participants

characteristic	Type	values
Age (years), mean (SD)	Men	33.5 (6.5)
	Women	31 (6.8)
BMI, mean (SD)	Men	25.58 (2.94)
	Women	24.32 (6.17)
Exercise	Almost daily	9
	A few times a week	19
	Once a week or fewer	7
Education	Primary school	1
	High school	7
	College	8
	University	20
Employment status	Working	27
	Unemployed	1
	Student	6
	Other	1

4 Dataset

PPG dataset used in this paper is a part of a health monitoring study [38]. During the study, the participants were asked to wear Samsung Gear Sport smartwatches, by which their vital signs, physical activity, and sleep were tracked continuously. The monitoring was performed under free-living conditions, where the participants engaged in their normal daily routines.

The recruitment and data collection took place in southern Finland between July and August 2019. The recruitment started with the students and staff members of the University of Turku. More recruiting was then done with snowball sampling, and in the end, 46 individuals were recruited. All of the participants were healthy individuals, and both males and females were present in equal numbers. Following exclusion criteria were used in the recruitment: (1) any restrictions using wearable devices at work, (2) restrictions regarding physical activity, (3) a diagnosed cardiovascular disease, and (4) symptoms of illness at recruitment time. Due to technical and practical issues, PPG signals from all 46 participants were not available, and data from 10 participants had to be excluded. Thus, PPG data from 36 participants were used in our analysis. Table 1 summarizes the background information of the participants.

All PPG signals were recorded with Samsung Gear Sport smartwatches [39]. The smartwatch has compact dimensions of 44.6 x 42.9 x 11.6 mm, and it weighs 67 grams with the strap. The smartwatch is waterproof, its battery lasts about 3 days, and includes a PPG sensor and a built-in inertial measurement unit. The device runs an open-source Tizen operating system, enabling customized data collection and data transmission.

For the data collection, the participants were asked to wear the smartwatches on their non-dominant hands. The watches were programmed to collect data for 24 hours at the sampling frequency of 20 Hz. We upsample the PPG signals to 100 Hz to include the tolerance distance in the peak detection (See section 5.3.2). The participants were also asked to send the collected data via Wi-Fi to our server using our Tizen app [38]. Our monitoring system is depicted in Figure 2, including the Samsung Gear sport smartwatch for data collection, a smartphone as a gateway layer for data transmission, and the cloud server.

This study was conducted following the ethical principles set by the Declaration of Helsinki and the Finnish Medical Research Act (No 488/1999). In addition, the University of Turku Ethics committee for

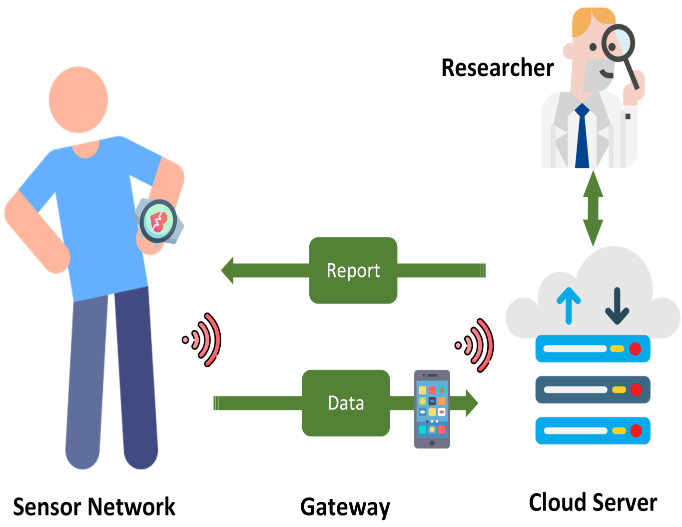

Figure 2: The monitoring system used for PPG collection.

Human Sciences gave a favorable statement (No 44/2019) of the study protocol. All study participants received both oral and written information about the study before their written consent was obtained. Study participation was entirely voluntary, and at all times, the participant had a right to withdraw from the study without giving any reason. At the end of the monitoring period, each participant was compensated with a 20 € gift card.

5 deep learning-based ppg peak detection

In this section, we present a deep-learning-based method designed for PPG peak detection. The method is trained using noisy PPG signals. In the following, we first describe the data preparation step, including a generator function to produce noisy PPG signals. We then present the proposed model architecture and peak extraction method. The data analysis pipeline is shown in Figure 3.

5.1 Data preparation

Data preparation generates noisy PPG signals with different SNRs to train and test the proposed model. The signals are generated using the available database, presented in Section 4. In this regard, we extract clean PPG signals and noise from the database. The collected PPG signals are nonstationary in terms of the noise level. In other words, the noise levels vary throughout the monitoring due to, for example, the user's hand movement. Hence, the signals are divided into (quasi)stationary segments, within which we assume the noise level is fixed. The length of the segments should be long enough to allow meaningful waveform analysis but short enough to ensure the segments are (quasi)stationary. Note that too short segments lead to low-resolution features. In our analysis, 15-second segments are selected.

The clean PPG signals are obtained using a PPG quality assessment technique, including five morphological features (i.e., spectral entropy, Shannon entropy, approximate entropy, kurtosis, and skewness) [40] and a Support Vector Machine method. We also obtain the systolic peak locations of the clean PPG signals using a derivative-based method [40]. Moreover, baseline wander and motion artifacts are extracted. Then, the clean signals (along with the peak locations) and noises are fed to a generator function.

5.1.1 Generator Function

A generator function is designed to create noisy PPG signals by aggregating clean PPG with noise. The noisy PPG signals are then utilized for training and testing the model. The generator function returns batches of normalized noisy PPG signals, their SNR values, and systolic peaks labels. Figure 4(a) shows a view of a generated PPG signal and its labeling vector.

The generator function includes 5 steps as follows:

Clean PPG signal selection: A 15-second window of clean PPG signal (X) is randomly selected.

Noise selection: A 15-second window of noise (N) is randomly selected. Note that our dataset noise includes baseline wander and motion artifacts.

Noisy PPG generation: A weighted arithmetic mean is utilized to create the noisy PPG signals:

$$S = w_X X + w_N N \tag{1}$$

where w_X and w_N are the weights of the clean PPG signal and noise, respectively. In our case, w_X is

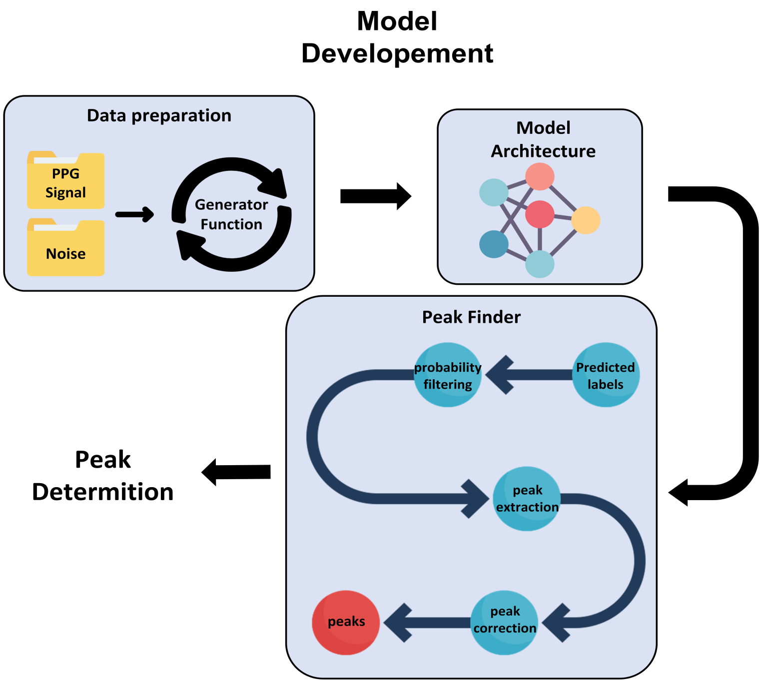

Figure 3: The proposed PPG peak detection method including data preparation, model architecture, and peak finder.

Algorithm 1 The generator function

Initialize:

win_size ← window size

batch_size ← number of batch size

$w_X \leftarrow 1$

while $i <$ batch_size **do**

$X \leftarrow$ select a window of the clean PPG signal randomly

clean peak ← Extract the corresponding peaks locations

$N \leftarrow$ select a random noise with same window size

$w_N \leftarrow$ a random number with uniform distribution (0,5)

$S \leftarrow w_X X + w_N N$

norm_sig ← normalize the noisy signal (i.e., S)

label ← create a binary label format for the noisy signal

SNR ← calculate the SNR

$i += 1$

end while

=0

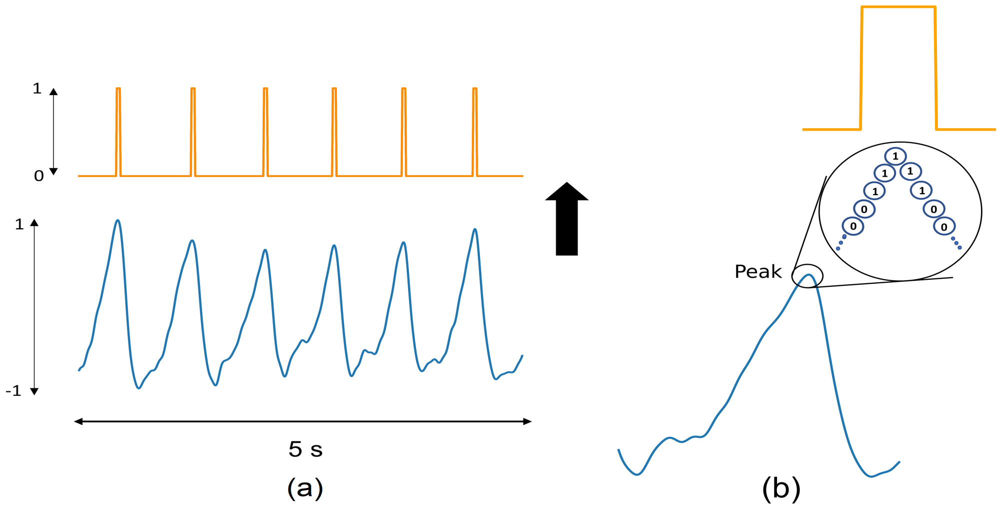

Figure 4: Schematic of labeling for PPG signal (a) a five-second PPG signal and its labeling vector. "1" indicates the peak's location (b) Five labels are set to "1" for each peak.

1 while w_N is a random number with uniform distribution (0,5). Therefore, PPG signals with different noise levels are constructed. Then, the signals are normalized to [-1, 1] to be used for training and testing our model.

Labels extraction: A binary format is used for labeling the systolic peaks in the constructed PPG signal. In this labeling, "1" corresponds to the peak locations, whereas the rest of the signal is labeled as "0." Moreover, a slightly balanced "1" is added to the adjacent systolic peak points for making the model more robust against the false positive. In other words, despite considering one point as the location of the peaks, five labels (i.e., peak, two preceding and two succeeding points) are set to "1" (see Figure 4). The use of five "1" instead of only one "1" in the labeling vector leads to more robust positive predictions. Therefore, it reduces the noise effect in identifying the peak's location. It should be noted that the label values are created according to the systolic peaks in the clean PPG signals (but not in the aggregated noisy PPG).

SNR extraction: The SNR is calculated for each constructed noisy PPG signal as follows:

$$SNR = 10 \log \frac{P_{Signal}}{P_{Noise}} \tag{2}$$

where P_{Signal} and P_{Noise} are the signal and noise powers, respectively. The procedure of generator function is also indicated in Algorithm 1.

5.2 Model architecture

To detect PPG peaks, we develop a CNN architecture with dilated convolutions, also known as atrous convolutions (or convolution with holes). Using dilated convolution instead of the regular one will result in a larger receptive field with the same amount of trainable parameters. This is achieved by inserting holes into the filter, i.e., some of the inputs are skipped as indicated in Figure 5. Dilation rate controls the amount of skipping, and filters with higher dilation rates have more holes. Dilated convolution with a dilation rate of 1 is a particular case that equals to a standard convolution. Dilated convolutions were utilized first time in efficient wavelet decomposition [41]. Later, they have been successfully utilized in different deep learning applications, such as semantic image segmentation [42], [43] and audio generation [44].

In our CNN model architecture, dilated convolutional layers are stacked, and their dilation rate is doubled at every layer. This approach results in vast receptive fields even with few layers (see Figure 6), which is computationally very efficient. Moreover, the input resolution is retained through the network.

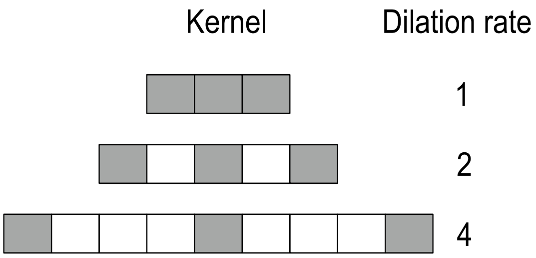

Figure 5: Receptive field of the kernel increases when dilation rate is increased. Grey denotes kernel weights while white indicates skipped inputs.

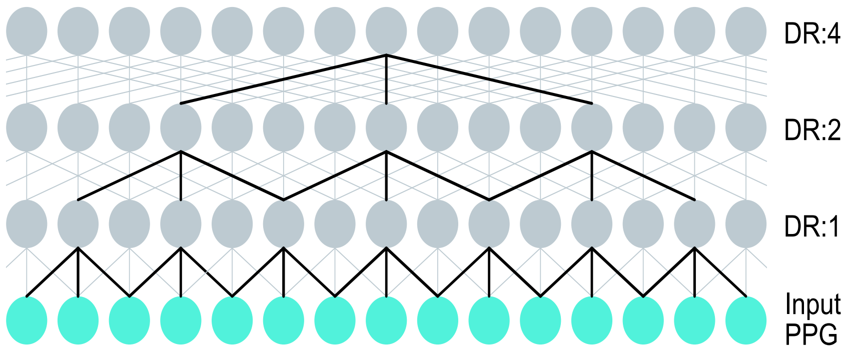

Figure 6: The receptive field of a neuron in a three-layer dilated convolution network is illustrated with bold lines. Note how the dilation rate (DR) is doubled at every layer. Our model is deeper than this illustration as it contains four additional layers.

In contrast with our method, other existing methods that expand the receptive field like strided convolutions (stride larger than 1) or pooling layers reduce the spatial resolution [45]. Stacking dilated causal convolutional layers and simultaneously increasing dilation rate was first proposed by Oord *et al.* [44] in part of their wavenet architecture for audio generation. Our model is architecturally simpler, as we use a feedforward structure without any residual or skip connections. We also do not enforce causality. Therefore, receptive fields of the neurons in our model can contain both preceding and succeeding information, which will allow our model to make more accurate predictions.

Our model is fully convolutional, and it is a stack of 7 1D convolutional layers as indicated in Figure 7. The input resolution of 1500 time steps is retained through the model. The kernel size is also 3 for every layer. The model makes sequence to sequence mapping. It produces a probability value for every time step. The probability value indicates how likely a signal point is a systolic peak. Two PPG examples with the probability values (i.e., the CNN model predictions) are shown in Figure 8. The dilatation rate is 1 in the first convolutional layer, and it is doubled at every following layer, reaching 64 at the final convolutional layer. This network structure results to a wide receptive field of 255 time steps for a neuron in a final classification layer. To keep our model compact, we slowly increase the number of filters as the network gets deeper. The first convolutional layer contains just four filters, while the second to last convolutional layer contains 32 filters. The final convolutional layer does the binary classification; therefore, it has only one filter. It uses sigmoid as an activation function while all preceding layers use exponential linear unit [46] as activation function. Moreover, we chose Adam [47] as an optimizer and binary cross-entropy as loss function. The proposed model is very small since it only has 3169 trainable parameters.

5.3 Peak finder

We develop a Wrapper function to extract the locations of the precise peaks from the model predictions, provided by the CNN model. Moreover, the Wrapper function detects and removes false peaks from the

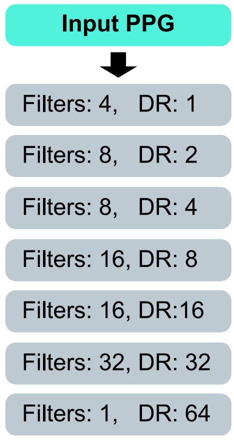

Figure 7: Our model is a fully convolutional neural network with seven layers. Dilatation rate (DR) is doubled at every layer while the number of filters is slowly increased with depth.

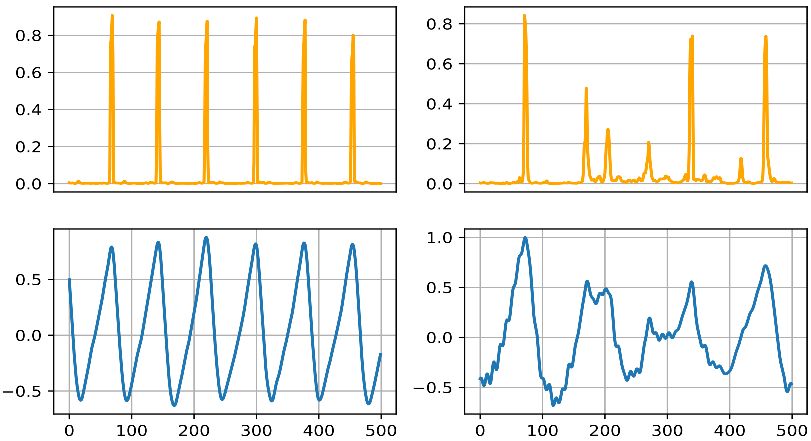

Figure 8: Two examples of inputs (i.e., PPG with different noise levels) and the model predictions. The lower row shows the two inputs, and the upper row includes the two labeling vectors predicted by model.

model predictions. Mainly, the function performs the three following tasks:

1. Removing the peaks with low probability predictions.
2. Extracting the precise peak locations within the predicted values.
3. Discarding false peaks in the predictions.

In the following, we describe the three tasks of the Wrapper function in more details.

5.3.1 Low probability signal removal

In the first step, predictions with low probability are discarded using a threshold value. A labeling vector –which each time step indicates one probability value between 0 and 1– is fed to the wrapper function. Then, a local threshold filter is applied to the predicted time steps, and the time steps below the defined probability value are filtered out. The threshold is chosen empirically after a considerable number of predicted time steps evaluation. Decreasing the probability threshold improves the recall but reduces the precision.

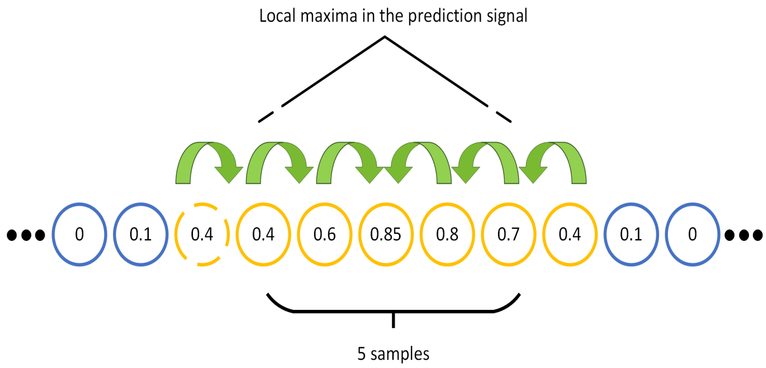

Figure 9: The procedure of selecting prediction that are above given threshold.

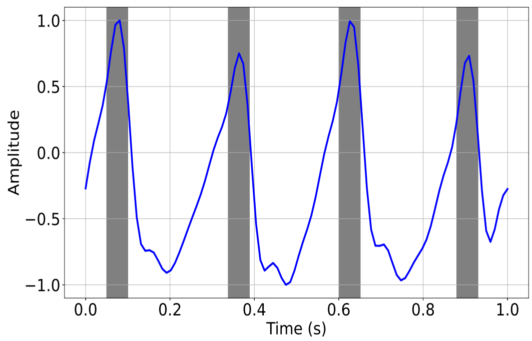

Figure 10: Tolerance distance (50 ms) is shaded in the figure. If the peak is detected within this range it is considered as true peak.

5.3.2 Peak Extraction

We use a local maximum finder to determine the exact peak's location. For this purpose, we design a searching function to find the five samples segment that has the higher value of the probability within the model predictions. In each five samples segment of the predicted time steps, the index of the higher probability is chosen as the location of the peak. Moreover, if there are two same probability values in the selected segment, the first probability value is chosen, and the corresponding index is extracted as the location of the peak. Figure 9 illustrates a segment of model prediction with its corresponding probability values. As shown in Figure 9, seven samples are above the threshold. In this example, the function finds the sample with the highest probability (i.e., 0.85) by comparing the neighbor points.

We produce a balance labeling vector for the noisy PPG signals, as instructed in Section 5. This idea helps our model to achieve higher precision while maintaining a lower tolerance distance. In other words, in the data generation stage, we introduce a new method for labeling. The method was generating a series of five "1" instead of only one "1" as the location of the peak. This means that if the algorithm finds a peak in the peak detection phase, there might be a time difference between the exact peak location and the detected peak. This time difference is introduced as tolerance distance. Figure 10 shows a segment of PPG signal and the defined tolerance distance with gray shaded rectangulars. In the proposed method, the tolerance distance is 50 ms, which is smaller than the tolerance distance (i.e., 88 ms) in other studies in the literature [48], [37].

5.3.3 Peak correction

In the third step, too-close peaks are discarded. Ventricular depolarization cannot occur in the refractory period despite the presence of stimuli. Therefore, no peak is presented in PPG signals during the refractory period after a peak. Our analysis assumes that the maximum heart rate is 200 beats per

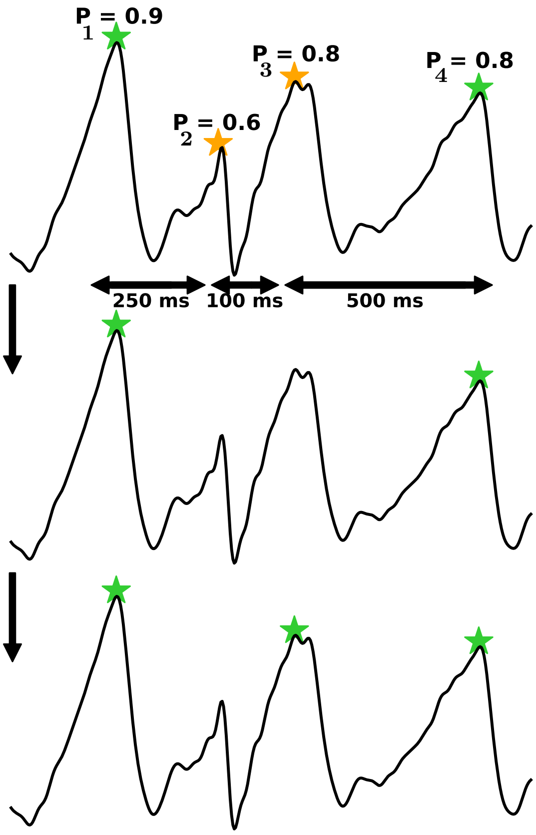

Figure 11: Filtering the unnaturally close peaks in the peak correction step. The upper figure shows all of the detected peaks. The middle plot illustrates the removal of the peaks that are within the threshold distance. The lowermost figure shows the peak with the highest probability is added back to the final set of accepted peaks.

minute, and accordingly, the minimum distance between two successive peaks is 300 milliseconds. This step is necessary when we aim to maximize the recall while a low-value probability threshold is defined.

Accordingly, the PPG peaks within a distance less than the threshold (i.e., 300 ms) are considered as false peaks. In this regard, we add a peak into the false-peak list if the distance with its preceding peak is less than 300 ms. Then, the false-peak list is sorted based on the peaks probabilities. In the next step, we select the highest peaks probability in the false-peak list and add it to the peak list. Then, we calculate the distance with its preceding peak. If the distance would be larger than the threshold, it is chosen as a peak; otherwise, it is removed. We repeat this step until the false-peak list is empty. For clarity, let us take an example of the PPG peak correction. Four systolic peaks are indicated in Figure 11. In the first round, we calculate the distance between each peak with its preceding peak. As shown in the figure, the distance between the first peak (P_1) and the second peak (P_2) is 250 ms. Therefore, P_2 is added to the false-peak list. Likewise, P_3 is added to the false-peak list. In the next step, we sort these false peaks based on their probabilities. Then, we start with the highest probability (i.e. P_2) and calculated its distance with P_1. The distance is 350 ms, which is above the threshold. Hence, P_3 is considered as a systolic peak. In the next round, we choose P_2 and follow the same procedure. As the distance between P_2 and P_1 is less than the threshold, P_2 is not a systolic peak and is removed from the false-peak list.

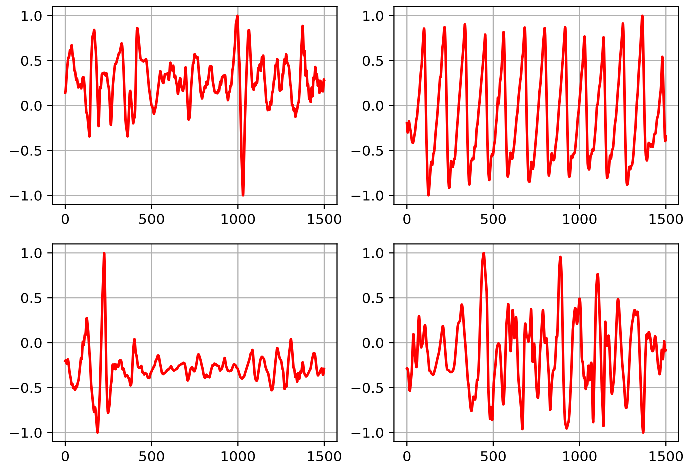

Figure 12: Examples of PPG data with different noise levels used in the training phase. The upper right figure illustrates a low noise PPG signal, while the other examples contain different types of noises such as motion artifact and baseline wander.

6 Evaluation and Results

We evaluate the proposed method using the PPG data collected via the Samsung smartwatches in free-living conditions. The evaluation includes the data of 36 healthy individuals. The model generalization is an important factor, which should be taken into consideration. We validate the performance of the proposed method by implementing an inter-patient test, in which training and testing data are selected from separate individuals. In this regard, the PPG data of 26 participants (i.e., 9 600 000 15-second segments) are utilized for the training phase. We train the proposed model using 1) the noisy PPG signals constructed via the generator function and 2) their true labeling vectors. For the testing phase, the data of the rest 10 participants (i.e., 35800 15-second segments) are selected. We separate the users to avoid any data leakage between the model training and testing. Similar to the training phase, the generator function is utilized to create noisy PPG segments. The test PPG signals are fed to the model, and the labeling vectors are estimated. Then, the method's performance is assessed by comparing the estimated labeling vectors with the true labeling vectors.

In our experiments, we used a Linux machine with AMD Ryzen Threadripper 2920X 12-Core processor, NVIDIA TITAN RTX GPU (24 GB memory), and 126 GB RAM. We use Tensorflow (v2) deep learning framework with high-level Keras API to construct our model. A batch size of 800 and 200 epochs, where the number of steps per epoch was 60, was selected for model training. In the training data, the range of SNR is from -2.5 to 47.5 dB (complete noisy to noise-free signal). The data are clustered into 10 ranges with the step of 5 dB. This balancing prevents the network from being over-learned for a specific SNR value. A total number of 9 600 000 segments were used for the training phase (90% training and 10% validation). Figure 12 indicated four examples of 15-seconds noisy PPG signals, with different noise levels, used in the training phase. The method was implemented using Tensorflow [49], Keras [50], and SciPy [51] in Python.

In addition to the proposed method, we implement four exiting methods for PPG peak detection. First, Elgendi *et al.* method [52] is performed, in which a dynamic threshold and two event-related moving average methods are utilized. Second, we utilize Van Gent *et al.* method [30] as an adaptive threshold method. Van Gent *et al.* [30] uses an adaptive threshold along with a moving average on both sides of each sample. Third, Kuntamalla *et al.* [8] method is implemented to estimate PPG peaks using an adaptive threshold, which is empirically set to 0.35. Fourth, Chakraborty *et al.* [33] as a transform-based method is used to estimate the peaks' locations using a Hilbert transform. It should be noted that For the Elgendi and Van Gent methods, we use the versions that are implemented in Neurokit [53] and Heartpy [54] Python packages.

6.1 Evaluation measures

A beat-to-beat comparison was made between the detection results and the reference test set label to evaluate the algorithm in terms of accuracy. In the comparison, true-positive (TP) is when the PPG peaks are detected correctly, false-negative (FN) is when the method fails to detect a peak, and false-positive (FP) is when the algorithm detects, e.g., noise as a peak. Then, the performance of the proposed method is assessed by calculating precision, recall, and F1-score as follows [32]:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{3}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\begin{aligned} \text{F1-score} &= 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \\ &= 2 \times \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \end{aligned} \tag{5}$$

6.2 Test set results

Our proposed method is evaluated using the test dataset created by the generator function. The function generates 100 Hz noisy PPG signals along with the SNR values and the corresponding labeling vectors. 35800 noisy PPG signals with a balanced range of SNR are used for the testing. The SNR values are between -2.5 and 47.5 dB in our evaluation. The signals are divided into 5-dB-SNR groups. Then, the performance of the methods is investigated for each group.

Figure 13 shows a PPG segment with different peak detection results. The SNR is 6.82 dB. The vertical dash lines show the true peaks, and the markers indicates the estimated peaks. Our method misses one systolic peak in second 10.8. However, the other methods miss several peaks and detect false peaks as systolic peaks. The Kuntamalla method had the worst performance in this example.

The performance of the models for different SNR groups are shown in Figure 14. A quantitative comparison is also presented in Table 2. Figure 14(a) illustrates the methods' precision. All the methods except the Kuntamalla method obtain equal precision value (i.e., 97%) when the SNR is above 42.5 dB. However, the precision values drop when the SNR values decrease. For example, in SNR 45 to 25 dB, the precision for the proposed method, Elgendi, Van Gent, Chakraborty, and Kuntamalla decreased by 15%, 18%, 24%, 19%, and 20%, respectively. As indicated, the proposed deep learning-based method outperforms the existing methods. The results show that the false positive in the proposed method is lower compared to the other methods. Therefore, our method detects fewer false peaks as systolic peaks in the noisy PPG signals.

Figure 14(b) indicates the methods' Recall values. The figure shows that all the methods perform well in noise-free conditions, i.e., almost 96% recall. As indicated, there are decreasing trends in the recall values when the SNR decreases. The falling trends are more intense in lower SNRs. With the least SNR, the difference between our method and the other methods reaches the highest value. As presented in Table 2 in SNR 0 dB, the differences between our method and the other methods are 7% (Elgendi), 8% (Van Gent), 12% (Kuntamalla), and 18% (Chakraborty). Similar to the precision, our method performs better in all SNR groups compared to the existing methods. The recall values show that our method obtains lower false negatives. Therefore, our method is more successful in detecting the true peaks.

Finally, the methods' F1-score values are illustrated In Figure 14(c). When the SNR values drop, the F1-score of the proposed method decrease with a smaller slope compared to the existing methods. The difference is bigger with lower SNR values. For example, as shown in Table 2, the F1-score values in 0 dB SNR are 0.52, 0.46, 0.43, 0.40, and 0.38 for our method, Elgendi, Van Gent, Chakraborty, and Kuntamalla, respectively. Consequently, the proposed method has the best performance with all the SNR groups, particularly when the SNR values are small. The method is more robust against noises and could better discriminate between the systolic and noise peaks.

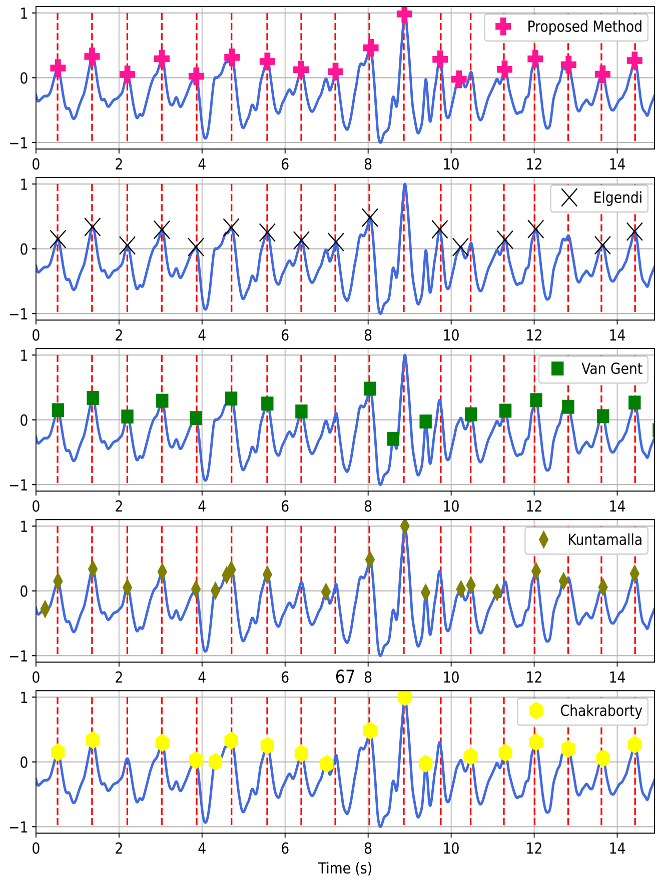

Figure 13: A PPG signal segment with 6.82 dB SNR and the peak detection results obtained by the methods. The vertical dash lines are true peak. The markers show the positions of the peaks detected by the methods.

Computation time: In addition to the accuracy assessment, we evaluate the computation time of the testing phase. We repeat the experiments 100 times and calculate the computation time of the methods. The average values and standard errors are indicated in Table 3. The Elgendi method (including rule-based steps) has the lowest execution time: i.e., 0.75 ms. The execution time of the proposed deep learning method is 1.081 ms, on average, which is lower than the processing time of the Van Gent, Chakraborty, and Kuntamalla methods (i.e., 8.55 ms, 2.48 ms, and 2.55 ms respectively).

6.3 Limitations and future work

The dataset used in this paper was limited to healthy participants. However, other studies [34] indicated that arrhythmias –such as premature atrial contraction, premature ventricle contraction, and atrial fibrillation– might affect the accuracy of peak detection methods. The method's performance should be investigated with the data of non-healthy individuals to address the lack of generalizability of the results.

Moreover, our evaluation was restricted to one dataset collected during free-living conditions using Samsung Gear Sport smartwatches. The method performance should be evaluated with different physical activities. In our future work, we intend to validate our method with other databases, such as [55], [56], in which the users are engaged more in intense physical activities such as cycling and running.

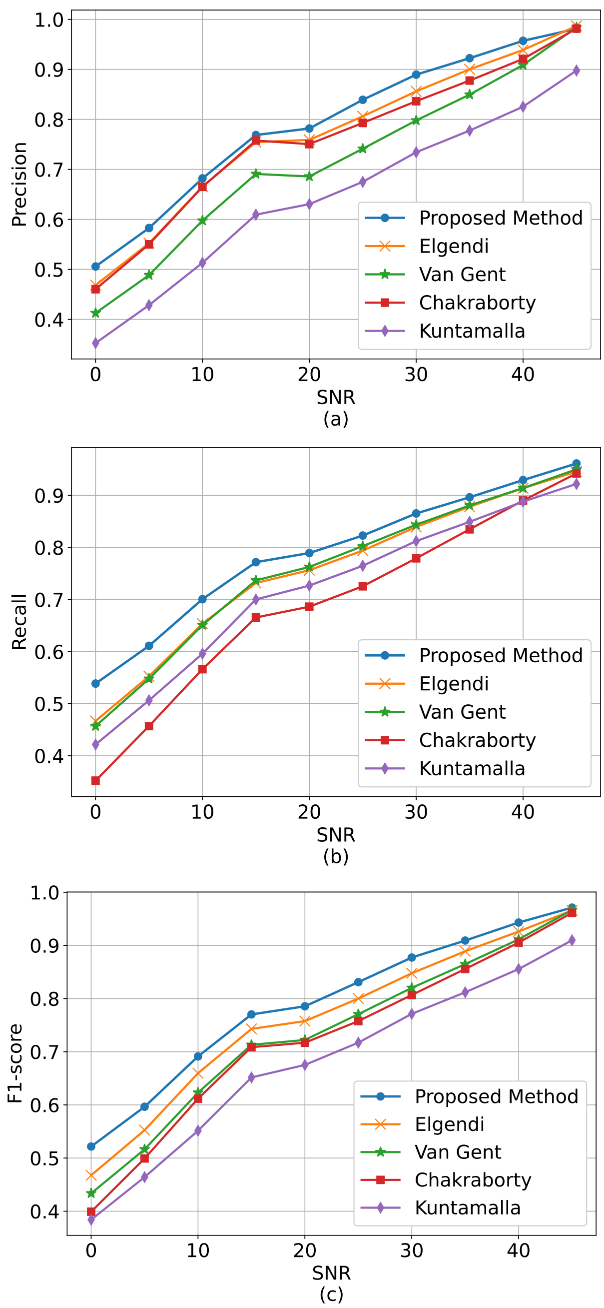

Figure 14: Performance comparison between the methods. (a) precision (b) recall (c) F1-score at different noise levels.

7 Conclusion

In this paper, we presented a robust CNN-based peak detection for PPG signals with different noise levels. The proposed method included three phases. A generator function was introduced in the first phase, combining the PPG records with different noise levels. In the second phase, a dilated CNN was proposed. The use of dilated convolutions provided a large receptive field, which enhanced the efficiency of time series processing with CNNs. In the third phase, a wrapper function was implemented to detect the location of the PPG signals. After predicting the peaks, a filtering function was used to remove the

Table 2: Performance comparison between the proposed method and the other existing methods

SNR	Proposed method			Elgendi			Van Gent			Chakraborty			Kuntamalla		
(dB)	prec.	recall	F1	prec.	recall	F1	prec.	recall	F1	prec.	recall	F1	prec.	recall	F1
45	**0.98**	**0.96**	**0.97**	**0.98**	0.94	0.96	**0.98**	0.95	0.96	**0.98**	0.94	0.96	0.89	0.92	0.90
40	**0.95**	**0.92**	**0.94**	0.93	0.90	0.92	0.91	0.91	0.91	0.92	0.88	0.90	0.82	0.88	0.85
35	**0.93**	**0.89**	**0.91**	0.90	0.88	0.89	0.86	0.88	0.87	0.88	0.83	0.85	0.78	0.85	0.81
30	**0.89**	**0.87**	**0.88**	0.87	0.85	0.86	0.81	0.85	0.83	0.85	0.79	0.82	0.74	0.82	0.78
25	**0.83**	**0.82**	**0.83**	0.80	0.79	0.80	0.74	0.80	0.77	0.79	0.73	0.76	0.69	0.77	0.73
20	**0.78**	**0.78**	**0.78**	0.76	0.75	0.76	0.69	0.76	0.72	0.75	0.68	0.71	0.63	0.73	0.67
15	**0.76**	**0.77**	**0.76**	0.75	0.73	0.74	0.68	0.73	0.71	0.75	0.66	0.70	0.60	0.69	0.64
10	**0.68**	**0.70**	**0.69**	0.66	0.66	0.66	0.60	0.66	0.63	0.67	0.58	0.62	0.52	0.61	0.56
5	**0.58**	**0.62**	**0.60**	0.55	0.56	0.56	0.49	0.55	0.52	0.55	0.46	0.50	0.42	0.50	0.46
0	**0.50**	**0.53**	**0.52**	0.46	0.46	0.46	0.42	0.45	0.43	0.46	0.35	0.40	0.35	0.41	0.38
Overall	**0.80**	**0.80**	**0.80**	0.78	0.76	0.77	0.72	0.77	0.74	0.78	0.76	0.77	0.65	0.73	0.69

*The corresponding PPG records with the highest precision, recall, or F1 scores (in each row) are presented in bold type.

*The number of signals analysed for each SNR range are 3580 .

Table 3: Average processing time comparison of the proposed method and other existing methods

Method	Proposed method	Elgendi	Van Gent	Chakraborty	Kuntamalla
Time (ms)	1.081± 0.31	0.75± 0.31	8.55± 0.73	2.48± 0.50	2.55± 0.53

false peaks. We evaluate the proposed method using the PPG data collected via wearable devices under free-living conditions. Our method was compared with 4 existing PPG peak detection methods. The performance of the methods were similar with noise-free PPG. However, our method exhibited higher accuracy when the noise level increased. We showed that the average F1-score of the proposed method was 80%, while Elgendi, Van Gent, Chakraborty, and Kuntamalla methods obtained 77%, 74%, 77%, and 69%, respectively. Our results indicated that the proposed PPG peak detection method was more successful in terms of recall and precision in a noisy environment.

Acknowledgment

The authors would like to thank Fatemeh Sarhaddi, Anna Axelin, Hannakaisa Niela-Vilen, Elisa Lankinen, Mohsen Saei Dehghan, Bushra Zafar, and Henrika Merenlehto for contributing to the data collection.

Funding

This research received support from the Academy of Finland through the SLIM Project (grant numbers 316810 and 316811) and the U.S. National Science Foundation (NSF) through the UNITE Project (grant number SCC CNS-1831918.)

References

[1] F. Firouzi, A. M. Rahmani, K. Mankodiya, M. Badaroglu, G. Merrett, P. Wong, B. Farahani, Internet-of-things and big data for smarter healthcare: From device to architecture, applications and analytics, Future Generation Computer Systems 78 (2018) 583–586. `doi:https://doi.org/10.1016/j.future.2017.09.016`.

[2] R. Mieronkoski, I. Azimi, A. M. Rahmani, R. Aantaa, V. Terävä, P. Liljeberg, S. Salanterä, The internet of things for basic nursing care—a scoping review, International journal of nursing studies 69 (2017) 78–90.

[3] J. Allen, Photoplethysmography and its application in clinical physiological measurement, Physiological measurement 28 (3) (2007) R1.

[4] S. Majumder, T. Mondal, M. J. Deen, Wearable sensors for remote health monitoring, Sensors 17 (1) (2017) 130.

[5] M. L. for Computational Physiology (2021), Physionet databases, [Online]. `https://physionet.org/about/database/`.

[6] M. Hahn, An adaptive ssf-based pulse peak detection algorithm for heart rate variability analysis in home healthcare environments, International Conference on Ubiquitous Healthcare (2010) 70–71.

[7] N. Paradkar, S. R. Chowdhury, Primary study for detection of arterial blood pressure waveform components, 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (2015) 1959–1962.

[8] S. Kuntamalla, L. R. G. Reddy, An efficient and automatic systolic peak detection algorithm for photoplethysmographic signals, International Journal of Computer Applications 97 (19).

[9] V. Ostojić, T. Lončar-Turukalo, D. Bajić, Empirical mode decomposition based real-time blood pressure delineation and quality assessment, Computing in Cardiology (2013) 221–224.

[10] D.-G. Jang, U. Farooq, S.-H. Park, M. Hahn, A robust method for pulse peak determination in a digital volume pulse waveform with a wandering baseline, IEEE transactions on biomedical circuits and systems 8 (5) (2014) 729–737.

[11] D.-G. Jang, S. Park, M. Hahn, S. hun Park, A real-time pulse peak detection algorithm for the photoplethysmogram, International Journal of Electronics and Electrical Engineering (2014) 45–49.

[12] T. Bhowmik, J. Dey, V. N. Tiwari, A novel method for accurate estimation of hrv from smartwatch PPG signals, 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (2017) 109–112.

[13] B. R. Ferro, A. R. Aguilera, R. F. de la Vara Prieto, Automated detection of the onset and systolic peak in the pulse wave using hilbert transform, Biomedical Signal Processing and Control 20 (2015) 78–84.

[14] M. S. Manikandan, K. Soman, A novel method for detecting R-peaks in electrocardiogram (ECG) signal, Biomedical Signal Processing and Control 7 (2) (2012) 118–128.

[15] K. Xu, X. Jiang, S. Lin, C. Dai, W. Chen, Stochastic modeling based nonlinear bayesian filtering for photoplethysmography denoising in wearable devices, IEEE Transactions on Industrial Informatics 16 (11) (2020) 7219–7230.

[16] E. K. Naeini, I. Azimi, A. M. Rahmani, P. Liljeberg, N. Dutt, A real-time PPG quality assessment approach for healthcare internet-of-things, Procedia Computer Science 151 (2019) 551–558.

[17] B. Sumukha, R. C. Kumar, S. S. Bharadwaj, K. George, Online peak detection in photoplethysmogram signals using sequential learning algorithm, 2017 International Joint Conference on Neural Networks (IJCNN) (2017) 1313–1320.

[18] A. Anzanpour, D. Amiri, I. Azimi, M. Levorato, N. Dutt, P. Liljeberg, A. M. Rahmani, Edge-assisted control for healthcare internet of things: A case study on PPG-based early warning score, ACM Transactions on Internet of Things 2 (1) (2020) 1–21.

[19] C. C. Aggarwal, et al., Neural networks and deep learning, Springer 10 (2018) 978–3.

[20] A. D. Orjuela-Cañón, D. Delisle-Rodríguez, A. López-Delis, R. F. de la Vara-Prieto, M. B. Cuadra-Sanz, Onset and peak pattern recognition on photoplethysmographic signals using neural networks, Iberoamerican Congress on Pattern Recognition (2013) 543–550.

[21] Y. Chen, M. Oyama-Higa, T. D. Pham, Identification of mental disorders by hidden markov modeling of photoplethysmograms, International Conference on Biomedical Informatics and Technology (2013) 29–39.

[22] A. Reiss, I. Indlekofer, P. Schmidt, K. Van Laerhoven, Deep PPG: Large-scale heart rate estimation with convolutional neural networks, Sensors 19 (14) (2019) 3079.

[23] T. Pereira, C. Ding, K. Gadhoumi, N. Tran, R. A. Colorado, K. Meisel, X. Hu, Deep learning approaches for plethysmography signal quality assessment in the presence of atrial fibrillation, Physiological measurement 40 (12) (2019) 125002.

[24] T. V. Tran, W.-Y. Chung, A robust algorithm for real-time peak detection of photoplethysmograms using a personal computer mouse, IEEE Sensors Journal 15 (8) (2015) 4651–4659.

[25] C. Fischer, B. Dömer, T. Wibmer, T. Penzel, An algorithm for real-time pulse waveform segmentation and artifact detection in photoplethysmograms, IEEE Journal of Biomedical and Health Informatics 21 (2) (2016) 372–381.

[26] I. Iliev, B. Nenova, I. Jekova, V. Krasteva, Algorithm for real-time pulse wave detection dedicated to non-invasive pulse sensing, Computing in Cardiology (2012) 777–780.

[27] R. R. F. de la Vara, D. D. Rodríguez, M. B. C. Sanz, A. S. Mengana, H. F. P. Quintero, et al., Algorithm for systolic peak detection of pulse wave, XXXVIII Conferencia Latinoamericana En Informatica (CLEI) (2012) 1–4.

[28] S. Vadrevu, M. S. Manikandan, Effective systolic peak detection algorithm using variational mode decomposition and center of gravity, IEEE Region 10 Conference (TENCON) (2016) 2711–2715.

[29] H. S. Shin, C. Lee, M. Lee, Adaptive threshold method for the peak detection of photoplethysmographic waveform, Computers in Biology and Medicine 39 (12) (2009) 1145–1152.

[30] P. van Gent, H. Farah, N. van Nes, B. van Arem, Analysing noisy driver physiology real-time using off-the-shelf sensors: Heart rate analysis software from the taking the fast lane project, Journal of Open Research Software 7 (1).

[31] P. van Gent, H. Farah, N. van Nes, B. van Arem, Heartpy: A novel heart rate algorithm for the analysis of noisy signals, Transportation research part F: traffic psychology and behaviour 66 (2019) 368–378.

[32] S. Vadrevu, M. S. Manikandan, A robust pulse onset and peak detection method for automated PPG signal analysis system, IEEE Transactions on Instrumentation and Measurement 68 (3) (2018) 807–817.

[33] A. Chakraborty, D. Sadhukhan, M. Mitra, A robust PPG onset and systolic peak detection algorithm based on hilbert transform, 2020 IEEE Calcutta Conference (CALCON) (2020) 176–180.

[34] D. Han, S. K. Bashar, J. Lazaro, E. Ding, C. Whitcomb, D. D. McManus, K. H. Chon, Smartwatch PPG peak detection method for sinus rhythm and cardiac arrhythmia, 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (2019) 4310–4313.

[35] Y. Xiang, Z. Lin, J. Meng, Automatic QRS complex detection using two-level convolutional neural network, Biomedical engineering online 17 (1) (2018) 1–17.

[36] H. Yang, M. Huang, Z. Cai, Y. Yao, C. Liu, A faster R CNN-based real-time QRS detector, 2019 Computing in Cardiology (CinC) (2019) Page–1.

[37] J. Laitala, M. Jiang, E. Syrjälä, E. K. Naeini, A. Airola, A. M. Rahmani, N. D. Dutt, P. Liljeberg, Robust ECG R-peak detection using LSTM, Proceedings of the 35th annual ACM symposium on applied computing (2020) 1104–1111.

[38] M. A. Mehrabadi, I. Azimi, F. Sarhaddi, A. Axelin, H. Niela-Vilén, S. Myllyntausta, S. Stenholm, N. Dutt, P. Liljeberg, A. M. Rahmani, Sleep tracking of a commercially available smart ring and smartwatch against medical-grade actigraphy in everyday settings: Instrument validation study, JMIR mHealth and uHealth 8 (11) (2020) e20465.

[39] Samsung gear sport, `https://www.samsung.com/global/galaxy/gear-sport`, accessed: 2010-09-30.

[40] A. Mahmoudzadeh, I. Azimi, A. M. Rahmani, P. Liljeberg, Lightweight photoplethysmography quality assessment for real-time iot-based health monitoring using unsupervised anomaly detection, Procedia Computer Science 184 (2021) 140–147.

[41] M. Holschneider, R. Kronland-Martinet, J. Morlet, P. Tchamitchian, A real-time algorithm for signal analysis with the help of the wavelet transform, in: Wavelets, Springer, 1990, pp. 286–297.

[42] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE transactions on pattern analysis and machine intelligence 40 (4) (2017) 834–848.

[43] L.-C. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation, arXiv preprint arXiv:1706.05587.

[44] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, Wavenet: A generative model for raw audio, arXiv preprint arXiv:1609.03499.

[45] Z. Wang, S. Ji, Smoothed dilated convolutions for improved dense prediction, Data Mining and Knowledge Discovery (2021) 1–27.

[46] D.-A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (elus), arXiv preprint arXiv:1511.07289.

[47] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.

[48] G. M. Friesen, T. C. Jannett, M. A. Jadallah, S. L. Yates, S. R. Quint, H. T. Nagle, A comparison of the noise sensitivity of nine QRS detection algorithms, IEEE Transactions on biomedical engineering 37 (1) (1990) 85–98.

[49] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., Tensorflow: A system for large-scale machine learning, 12th {USENIX} symposium on operating systems design and implementation ({OSDI} 16) (2016) 265–283, [Online] http://tensorflow.org/.

[50] F. Chollet, et al., keras, [Online] https://keras.io. (2015).

[51] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, et al., Scipy 1.0: fundamental algorithms for scientific computing in python, Nature methods 17 (3) (2020) 261–272.

[52] M. Elgendi, I. Norton, M. Brearley, D. Abbott, D. Schuurmans, Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions, PLoS One 8 (10) (2013) e76585.

[53] D. Makowski, T. Pham, Z. J. Lau, J. C. Brammer, F. Lespinasse, H. Pham, C. Schölzel, S. H. A. Chen, Neurokit2: A python toolbox for neurophysiological signal processing. behavior research methods, [Online]. https://doi.org/10.3758/s13428-020-01516-y (2021).

[54] P. van Gent, H. Farah, N. Nes, B. van Arem, Heart rate analysis for human factors: Development and validation of an open source toolkit for noisy naturalistic heart rate data, Proceedings of the 6th HUMANIST Conference (2018) 173–178.

[55] Z. Zhang, IEEE signal processing cup 2015: Heart rate monitoring during physical exercise using wrist-type photoplethysmographic (PPG) signals, [Online] https://sites.google.com/site/researchbyzhang/ieeespcup2015.

[56] D. Jarchi, A. J. Casson, Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals, [Online]. https://physionet.org/content/wrist/1.0.0/.