

# An Accurate Non-accelerometer-based PPG Motion Artifact Removal Technique using CycleGAN

AMIR HOSEIN AFANDIZADEH ZARGARI\*, SEYED AMIR HOSSEIN AQAJARI\*, HADI KHODABANDEH\*, AMIR M. RAHMANI, and FADI KURDAHI, University of California, Irvine, USA

A photoplethysmography (PPG) is an uncomplicated and inexpensive optical technique widely used in the healthcare domain to extract valuable health-related information, e.g., heart rate variability, blood pressure, and respiration rate. PPG signals can easily be collected continuously and remotely using portable wearable devices. However, these measuring devices are vulnerable to motion artifacts caused by daily life activities. The most common ways to eliminate motion artifacts use extra accelerometer sensors, which suffer from two limitations: i) high power consumption and ii) the need to integrate an accelerometer sensor in a wearable device (which is not required in certain wearables). This paper proposes a low-power non-accelerometer-based PPG motion artifacts removal method outperforming the accuracy of the existing methods. We use Cycle Generative Adversarial Network to reconstruct clean PPG signals from noisy PPG signals. Our novel machine-learning-based technique achieves 9.5 times improvement in motion artifact removal compared to the state-of-the-art without using extra sensors such as an accelerometer, which leads to 45% improvement in energy efficiency.

CCS Concepts: • **Applied computing** → **Health informatics**; • **Computing methodologies** → **Machine learning**.

Additional Key Words and Phrases: Machine Learning, Deep Generative models, Cycle GAN, PPG Signals, Motion Artifacts removal, Noise removal

## 1 INTRODUCTION

A photoplethysmography (PPG) is a simple, low-cost, and convenient optical technique used for detecting volumetric blood changes in the microvascular bed of target tissue [4]. Valuable health-related information can be extracted from PPG signals such as heart rate and heart rate variability.

Nowadays, PPG signals can easily be collected continuously and remotely using inexpensive, convenient, and portable wearable devices (e.g., smartwatches, rings, etc.) which makes them a suitable source in wellness applications in everyday life. However, PPG signals collected from portable wearable devices in everyday settings are often measured when a user is engaged with different kinds of activities and therefore are distorted by motion artifacts. The signal with a low signal-to-noise ratio leads to inaccurate vital signs extraction which may risk life-threatening consequences for healthcare applications. There exists a variety of methods to detect and remove motion artifacts from PPG signals. The majority of the works related to the detection and filtering of motion artifacts in PPG signals can reside in three categories: (1) non-acceleration based, (2) using synthetic reference data, and (3) using acceleration data.

The non-acceleration based methods do not require any extra accelerometer sensor for motion artifact detection and removal. In existing works, these approaches utilize certain statistical methods due to the fact that some statistical parameters such as skewness and kurtosis will remain unchanged regardless of the presence of the

\*First three authors contributed equally to this research.

Authors' address: Amir Hosein Afandizadeh Zargari, amir.zargari@uci.edu; Seyed Amir Hossein Aqajari, saqajari@uci.edu; Hadi Khodabandeh, khodabandeh@uci.edu; Amir M. Rahmani, a.rahmani@uci.edu; Fadi Kurdahi, kurdahi@uci.edu, University of California, Irvine, P.O. Box 92617, Irvine, CA, USA, 43017-6221.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

© 2022 Copyright held by the owner/author(s).

2637-8051/2022/9-ART

<https://doi.org/10.1145/3563949>

noise. In [17], such statistical parameters are used to detect and remove the impure part of the signal due to motion artifacts. In [12], authors detect motion artifacts using a Variable Frequency Complex Demodulation (VFCDM) method. In this method, the PPG signal is normalized after applying a band-pass filter. Then, to detect motion artifacts, VFCDM distinguishes between the spectral characteristics of noise and clean signals. Then, due to a shift in the frequency, an unclean-marked signal is removed from the entire signal. Another method in this category is proposed in [25] that uses the Discrete Wavelet Transform (DWT) method.

In non-accelerometer based methods, the clean output signal is often shorter than the original signal, since unrecovered noisy data is removed from the signal. To mitigate this problem, a synthetic reference signal can be generated from the corrupted PPG signal. In [33], authors use Complex Empirical Mode Decomposition (CEMD) to generate signals. In [18], two PPG sensors are being used to generate a reference signal. One of the sensors is a few millimeters away from the skin, which only measures PPG during movements. First a band-pass filter is applied on both recorded signals; then, an adaptive filter is used to minimize the difference between two recorded signals.

Sensors are the most critical part of wearable sensing devices, and their sensitivity plays an important role [26, 30]. Often an accelerometer sensor is also embedded in wearable devices. To eliminate the effect of motion artifacts, acceleration data can be used as a reference signal. In [37], with the help of acceleration data, Singular Value Decomposition (SVD) is used for generating a reference signal for an adaptive filter. Then, the reference signal and PPG signal pass through an adaptive filter to remove motion artifacts. With a similar approach, authors in [39] use DC remover using another type of adaptive filter. Another method for motion artifact removal is proposed in [11] which follows three steps: (1) signals are windowed, (2) the output signal is filtered, and (3) a Hankel data matrix is constructed.

Even though using an accelerometer-based method increases the model's accuracy, it suffers from two limitations: i) high power consumption and ii) the need to integrate an accelerometer sensor in a wearable device (which is not required in certain wearables). To overcome these issues, machine learning techniques can be employed as an alternative method to remove noise and reconstruct clean signals [9, 40, 41]. Furthermore, machine learning techniques, proven to be useful in numerous research areas [15, 27–29, 36], are utilized in healthcare domain in processing of a variety of physiological signals such as PPG for data analysis purposes [5–8, 13, 22]. The aim of this paper is to propose a machine learning non-accelerometer-based PPG motion artifacts removal method which is low-power and can outperform the accuracy of the existing methods (even the accelerometer-based techniques). In recent studies, applying machine learning for image noise reduction has been investigated extensively. The most recent studies use deep generative models to reconstruct or generate clean images [14, 38]. In this paper, we propose a novel approach which converts noisy PPG signals to a proper visual representation and uses deep generative models to remove the motion artifacts. We use a Cycle Generative Adversarial Network (CycleGAN) [43] to reconstruct clean PPG signals from noisy PPG. CycleGAN is a novel and powerful technique in unsupervised learning, which targets learning the distribution of two given datasets to translate an individual input data from the first domain to a desired output data from the second domain. The advantages of CycleGAN over other existing image translation methods are i) it does not require the pairwise dataset, and ii) the augmentation in CycleGAN makes it practically more suitable for datasets with fewer images. Hence, we use CycleGAN to remove motion artifacts from noisy PPG signals and reconstruct the clean signals. Our experimental results demonstrate the superiority of our approach compared to the state-of-the-art with a 9.5 times improvement with approximately 45% improvement in the energy efficiency due to eliminating accelerometer sensors.

The rest of this paper is organized as follows. Section Methods introduces the employed dataset and our proposed pipeline architecture. In section Results we summarize the result obtained by our proposed method and compare our result with the state-of-the-art in motion artifact removal from PPG signals. Finally, in the Conclusion section we discuss the strengths and limitations of our method and we cover the future work.

## 2 METHODS

In this paper, we present an accurate non-accelerometer-based motion artifacts removal model from PPG signals. This model mainly consists of a module for artifact detection and another one for motion artifact removal. We present in Figure 1 the flow chart of our proposed model.

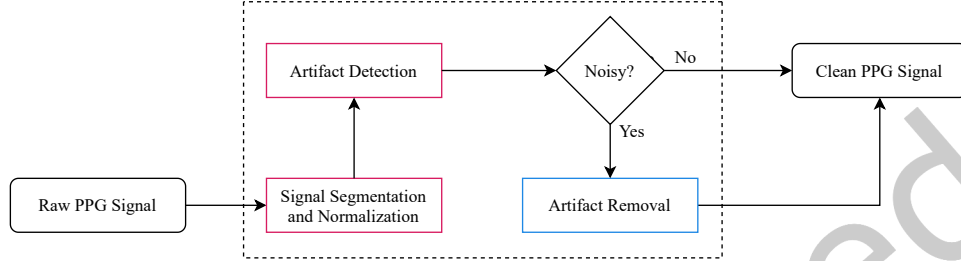


Fig. 1. Flowchart of the proposed method

The artifact removal module consists of sub-modules (Figure 2) that deliver the task of cleaning the input signal by transforming it into a two dimensional image and using CycleGAN to remove the two dimensional noise induced by the artifacts. Consequently, the clean image is transformed to a signal that is returned in the output. Each of these modules are discussed in detail in their corresponding sections.

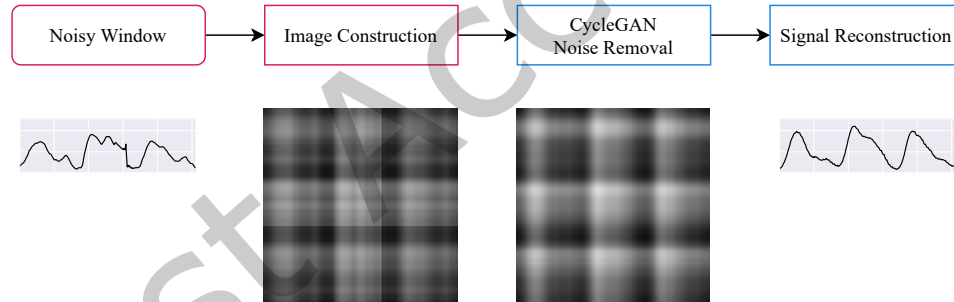


Fig. 2. Flowchart of the artifact removal module

In order to train this model, two datasets of PPG signals are required: one consisting of clean PPG signals and the other one containing noisy PPG signals (In the rest of this paper, by noisy PPG signals we are referring to PPG signals affected by motion artifact). The model's evaluation requires both clean and noisy signals to be taken from the same patient in the same period of time. However, recording such data is not feasible as patients are either performing an activity, which leads to recording a noisy signal or are in a steady-state, which produces a clean signal. For this reason, we simulate the noisy signal by adding data from an accelerometer to the clean signal. This is a common practice and has been used earlier in related work (e.g., [34]) to address this issue. This way, the effectiveness of the model can be evaluated efficiently by comparing the clean signal with the reconstructed output of the model on the derived noisy signal. In the following subsections, we explain the process of data collection for both clean and noisy datasets.

## 2.1 BIDMC Dataset

For the clean dataset, we use BIDMC dataset [31]. This dataset contains signals and numerics extracted from the much larger MIMIC II matched waveform database, along with manual breath annotations made from two annotators, using the impedance respiratory signal.

The original data was acquired from critically ill patients during hospital care at the Beth Israel Deaconess Medical Centre (Boston, MA, USA). Two annotators manually annotated individual breaths in each recording using the impedance respiratory signal. There are 53 recordings in the dataset, each being 8 minutes long and containing:

- Physiological signals, such as the PPG, impedance respiratory signal, and electrocardiogram (ECG) sampled at 125 Hz.
- Physiological parameters, such as the heart rate (HR), respiratory rate (RR), and blood oxygen saturation level (SpO2) sampled at 1 Hz.
- Fixed parameters, such as age and gender. The ages range from 19 to higher than 90. Also, out of 53 subjects in this dataset, 20 of them are males and 32 are females (one subject's sex is not determined).
- Manual annotations of breaths.

## 2.2 Data Collection

We conducted laboratory-based experiments to collect accelerometer data for generating noisy PPG signals. Each of these laboratory-based experiments consisted of 27 minutes of data. A total of 33 subjects participated in the laboratory-based experiments. The ages of the subjects ranged from 20 to 62, and 17 of them were males while 16 were females. In each experiment, subjects were asked to perform specific activities while the accelerometer data were collected from them using an Empatica E4 [2] wristband worn on their dominant hand. The Empatica E4 wristband is a research-grade wearable device that offers real-time physiological data acquisition, enabling researchers to conduct in-depth analysis and visualization. A recent research study detects and discriminates acute psychological stress (APS) in the presence of concurrent physical activity (PA) using the PPG and the Accelerometer data collected from Empatica E4 wristband [35]. Figure 3 shows our experimental procedure. Note that the accelerometer signals are only required for generating/emulating noisy PPG signals, and our proposed motion artifact removal method does not depend on having access to acceleration signals.

Rest	Finger Tapping	Finger Tapping	Rest	Waving	Waving	Rest	Shaking Hands	Shaking Hands
	Low Intensity	High Intensity		Low Intensity	High Intensity		Low Intensity	High Intensity
Activity 1			Activity 2			Activity 3		
Rest	Running Arm Swings	Running Arm Swings	Rest	Fist Open/Close	Fist Open/Close	Rest	3D Arm Movement	3D Arm Movement
	Low Intensity	High Intensity		Low Intensity	High Intensity		Low Intensity	High Intensity
Activity 4			Activity 5			Activity 6		

Fig. 3. Experimental procedure to collect accelerometer data.

According to Figure 3, each experiment consists of 6 different activities: (1) Finger Tapping, (2) Waving, (3) Shaking Hands, (4) Running Arm Swing, (5) Fist Opening and Closing, and (6) 3D Arm Movement. Each activity lasts 4 minutes in total, including two parts with two different movement intensities (low and high), each of

which lasts 2 minutes. Activity tasks are followed by a 30 seconds rest (R) period between them. During the rest periods, participants were asked to stop the previous activity and put both their arms on a table, and stay in a steady state. Accelerometer data collected during each of the activities were later used to model the motion artifact. We describe this in the next subsection.

### 2.3 Noisy PPG signal generation

To generate noisy PPG signals from clean PPG signals, we use accelerometer data collected in our study. Clean PPG signals are directly collected from the BIDMC dataset. Accelerometer data is taken at 32 Hz, thus we down-sample the clean signals to 32 Hz to ensure they are synchronized with the collected accelerometer data.

Empatica has an onboard MEMS type 3-axis accelerometer that measures the continuous gravitational force (g) applied to each of the three spatial dimensions (x, y, and z). The scale is limited to  $\pm 2g$ . Figure 4 shows an example of accelerometer data collected from E4.

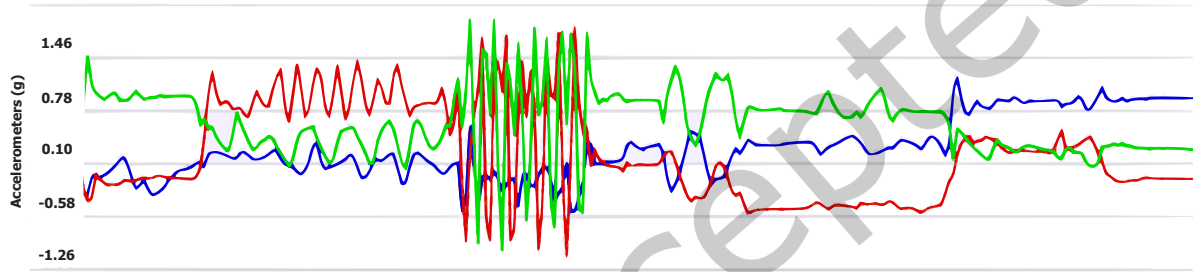


Fig. 4. An example of Accelerometer data from Connect, the subject moves into position, walks, runs, and then simulates the turning of a car's steering wheel. The dimensional axes are depicted in red, green and blue.

Along with the raw 3-dimensional acceleration data, Empatica also provides a moving average of the data. Every second, the following summation is calculated over the (32 samples) input received from the accelerometer sensor,

$$S = \sum_{t=1}^{32} \max(|\text{Acc}_x[t] - \text{Acc}_x[1]|, |\text{Acc}_y[t] - \text{Acc}_y[1]|, |\text{Acc}_z[t] - \text{Acc}_z[1]|) \quad (1)$$

where  $\text{Acc}_i[t]$  is the value of the accelerometer sensor (g) along the  $i$ -th dimension at time frame (sample)  $t$ , and  $\text{Acc}_i[1]$  is the first value of the accelerometer sensor (g) along the  $i$ -th dimension in the current window. The  $\max(a, b, c)$  function simply returns the maximum value among  $a$ ,  $b$ , and  $c$ . It is worth to mention that the values stored in the arrays  $\text{Acc}_x$ ,  $\text{Acc}_y$ , and  $\text{Acc}_z$  change after each window is processed.

Afterwards, the value of the moving average for the new window will be calculated based on this summation and the value of the moving average on the previous window,

$$\text{Avg}[w] = 0.9 \times \text{Avg}[w - 1] + 0.1 \times \frac{S}{32} \quad (2)$$

Figure 5 visualizes this moving average over the data.

This filtered output (Avg) is directly used as a model for motion artifacts in our study. To simulate the noisy PPG signals, we add this artifact model to a 2 minutes window of the clean PPG signals collected from the BIDMC dataset. We use 40 out of 53 signals in BIDMC directly as the clean dataset for training. Among these 40 signals, 20 are selected and augmented with the accelerometer data to construct the noisy dataset for training. The 13

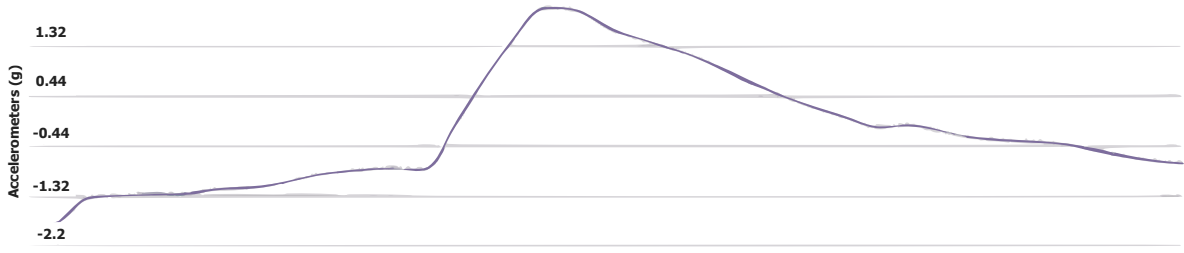


Fig. 5. The same data as Figure 4 is visualized using the moving average. From Connect, the subject moves into position, walks, runs, and then simulates the turning of a car's steering wheel. The dimensional axes are depicted in red, green and blue.

remaining BIDMC signals and accelerometer data were added together to form the clean and noisy datasets for testing. In the rest of this section we describe each part of the model introduced in Figure 1.

## 2.4 Noise Detection

To perform noise detection, first, the raw signal, which is downsampled to 32 Hz, is normalized by a linear transformation to map its values to the range (0, 1). This can be performed using a simple function as below:

$$\text{Sig}_{\text{norm}} = \frac{\text{Sig}_{\text{raw}} - \min(\text{Sig}_{\text{raw}})}{\max(\text{Sig}_{\text{raw}}) - \min(\text{Sig}_{\text{raw}})} \quad (3)$$

where  $\text{Sig}_{\text{raw}}$  is the raw signal and  $\text{Sig}_{\text{norm}}$  is the normalized output. Then, the normalized signal is divided into equal windows of size 256, which is the same window size we use for noise removal. These windows are then used as the input of the noise detection module to identify the noisy ones.

The similar type of machine learning network used in [42] can be employed as a noise detection system. To explain the network structure for the noise detection method (Table 1 and Figure 6), first, we use a 1D-convolutional layer with 70 initial random filters with a size of 10 to select the basic features of the input data and convert the matrix size from  $256 \times 1$  to  $247 \times 70$ . To extract more complex features from the data, another 1D-convolutional layer with the same filter size 10 is required. As the third layer, a pooling layer with a filter size of 3 is utilized. In this layer, a sliding window slides over the input of the layer and in each step, the maximum value of the window is applied to the other values. This layer converts a matrix size of  $238 \times 70$  to  $79 \times 70$ . To select additional complex features, another set of convolutional layers are used with a different filter size. This set is followed by two fully connected layers of sizes 32 and 16. Lastly, a dense layer of size 2 with a softmax activation would produce the probability of each class: clean and noisy. The maximum of these two probabilities would be identified as the result of the classification. The accuracy of our proposed binary classification method is 99%, which means that the system can almost always detect a noisy signal from a clean signal.

## 2.5 Noise Removal

In this section, we explore the reconstruction of noisy PPG signals using deep generative models. Once a noisy window is detected, it is sent to the noise removal module for further processing. First, the windows are transformed into 2-dimensional images, to exploit the power of existing image noise removal models, and then a

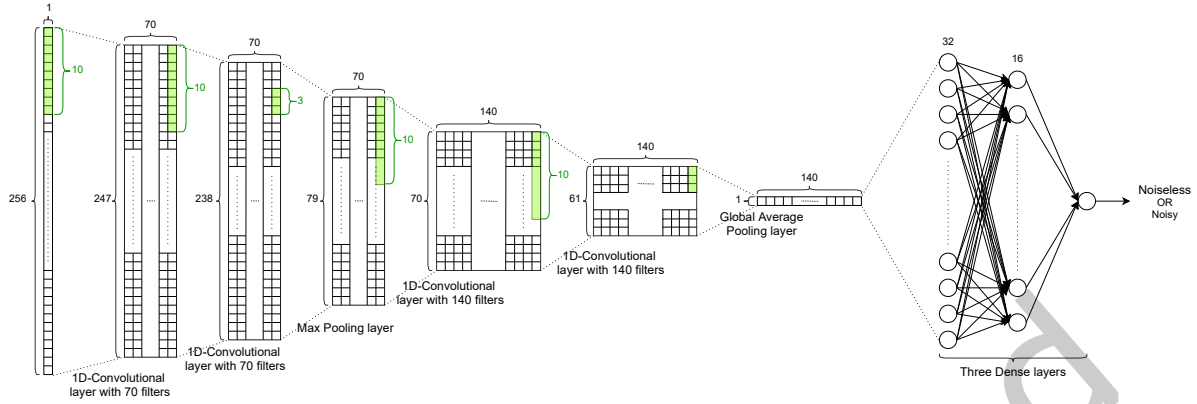


Fig. 6. The structure of the noise detection model.

Table 1. The layer configuration of the noise detection model.

Layer	Structure	Output
Conv1D+Relu	$70 \times 10$	$247 \times 70$
Conv1D+Relu	$70 \times 10$	$238 \times 70$
Max pooling 1D	3	$79 \times 70$
figure Conv1D+Relu	$140 \times 10$	$70 \times 140$
Conv1D+Relu	$140 \times 10$	$61 \times 140$
Global average pooling	N/A	140
Dense+Relu	128	32
Dense+Relu	16	16
Dense+Softmax	2	2

trained CycleGAN model is used to remove the noise induced by the motion artifact from these images. In the final step of the noise removal, the image transformation is reversed to obtain the clean output.

The transformation needs to provide visual features for unexpected changes in the signal so that the CycleGAN model would be able to distinguish and hence reconstruct the noisy parts. To extend the 1-dimensional noise on the signal into a 2-dimensional visual noise on the image, we apply the following transformation:

$$\text{Img}[i, j] = \text{floor}((\text{Sig}[i] + \text{Sig}[j]) \times 128) \quad (4)$$

where Sig is a normalized window of the signal, Img is the 2d array storing the grayscale image, and  $i$  and  $j$  are time frames in the window. Each pixel, i.e. each entry of Img, will then have a value between 0 and 255, representing a grayscale image. An example of such transformation is provided in Figure 7 for both clean and noisy signals. According to this figure, the noise effect is visually observable in these images.

Autoencoders and CycleGAN are two of the most powerful approaches for image translation. These methods have proven to be effective in the particular case of noise reduction. Autoencoders require the pairwise translation of every image in the dataset. In our case, clean and noisy signals are not captured simultaneously, and their quantity differs. CycleGAN, on the other hand, does not require the dataset to be pairwise. Also, the augmentation

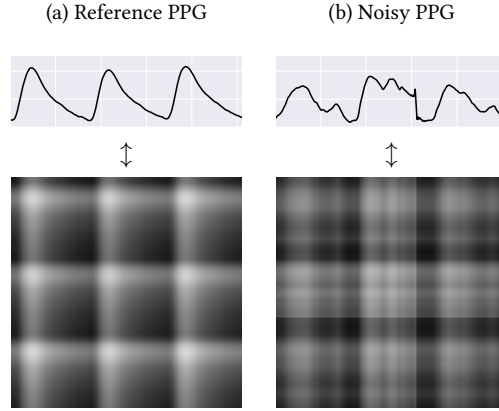


Fig. 7. An example of signal to image transformation.

in CycleGAN makes it practically more suitable for datasets with fewer images. Hence, we use CycleGAN to remove motion artifacts from noisy PPG signals and reconstruct the clean signals.

CycleGAN is a Generative Adversarial Network designed for the general purpose of image-to-image translation. CycleGAN architecture was first proposed by Zhu et al. in [43].

The GAN architecture consists of two networks: a generator network and a discriminator network. The generator network starts from a latent space as input and attempts to generate new data from the domain. The discriminator network aims to take the generated data as an input and predict whether it is from a dataset (real) or generated (fake). The generator is updated to generate more realistic data to better fool the discriminator, and the discriminator is updated to better detect generated data by the generator network.

The CycleGAN is an extension of the GAN architecture. In the CycleGAN, two generator networks and two discriminator networks are simultaneously trained. The generator network takes data from the first domain as an input and generates data for the second domain as an output. The other generator takes data from the second domain and generates the first domain data. The two discriminator networks are trained to determine how plausible the generated data are. Then the generator models are updated accordingly. This extension itself cannot guarantee that the learned function can translate an individual input into a desirable output. Therefore, the CycleGAN uses a cycle consistency as an additional extension to the model. The idea is that output data by the first generator can be used as input data to the second generator. Cycle consistency is encouraged in the CycleGAN by adding an additional loss to measure the difference between the generated output of the second generator and the original data (and vice versa). This guides the data generation process toward data translation.

In our CycleGAN architecture, we apply adversarial losses [16] to both mapping functions ( $G : X \rightarrow Y$  and  $F : Y \rightarrow X$ ). The objective of the mapping function  $G$  as a generator and its discriminator  $D_Y$  is expressed as below:

$$L_{GAN}(G, D_Y, X, Y) = E_{y \sim p_{data}(y)}[\log \log D_Y(y)] + E_{x \sim p_{data}(x)}[\log \log(1 - D_Y(G(x)))] \quad (5)$$

where the function  $G$  takes an input from domain  $X$  (e.g., noisy PPG signals), attempting to generate new data that look similar to data from domain  $Y$  (e.g., clean PPG signals). In the meantime,  $D_Y$  aims to determine whether its input is from the translated samples  $G(x)$  (e.g., reconstructed PPG signals) or the real samples from domain  $Y$ . A similar adversarial loss is defined for the mapping function  $F : Y \rightarrow X$  as  $L_{GAN}(F, D_X, Y, X)$ .



As discussed before, adversarial losses alone cannot guarantee that the learned function can map an individual input from domain  $X$  to the desired output from domain  $Y$ . In [43], the authors argue that to reduce the space of possible mapping functions even further, learned mapping functions ( $G$  and  $F$ ) need to be cycle-consistent. This means that the translation cycle needs to be able to translate back the input from domain  $X$  to the original image as  $X \rightarrow G(X) \rightarrow F(G(X)) \sim X$ . This is called forward cycle consistency. Similarly, backward cycle consistency is defined as:  $y \rightarrow F(y) \rightarrow G(F(y)) \sim y$ . This behavior is presented in our objective function as:

$$L_{\text{cyc}}(G, F) = E_{x \sim p_{\text{data}}(x)}[\|F(G(x)) - x\|_1] + E_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1] \quad (6)$$

Therefore, the final objective of CycleGAN architecture is defined as:

$$L(G, F, D_X, D_Y) = L_{\text{GAN}}(G, D_Y, X, Y) + L_{\text{GAN}}(F, D_X, Y, X) + \lambda L_{\text{cyc}}(G, F) \quad (7)$$

where  $\lambda$  controls the relative importance of the two objectives.

In Equation 7,  $G$  aims to minimize the objective while an adversary  $D$  attempts to maximize it. Therefore, our model aims to solve:

$$G^*, F^* = \text{argmin} L(G, F, D_X, D_Y) \quad (8)$$

The architecture of the generative networks is adopted from Johnson et al. [21]. This network contains four convolutions, several residual blocks [19], and two fractionally-strided convolutions with stride 0.5. For the discriminator networks, they use  $70 \times 70$  PathGANs [20, 23, 24].

After the CycleGAN is applied to the transformed image, the diagonal entries are used to retrieve the reconstructed signal.

$$\text{Sig}_{\text{rec}}[i] = \text{Img}[i, i]/256 \quad (9)$$

### 3 RESULTS

In this section, we assess the efficiency of our model based on the following measures: root mean square error (RMSE) and peak-to-peak error (PPE). A signal window size of 256 and an image size of 256 by 256 were used for all experimental purposes, and 25% of the data was assigned for validation. The noise detection module had an accuracy of 99%. The summary of the results for noise removal, including the improvement for each noise type and noise intensity, can be found in Table 2.

For each noise type, there are two entries in this table, one corresponding to the slow movement and the other one corresponding to the fast movement. The average S/N value for slow movements is 21.7dB, as provided in the table, while the average S/N value for fast movements is 14.0dB. For each of the measures, RMSE and PPE, we calculated the error between the generated signal and the reference signal as well as the error between the noisy signal and the reference signal in order to observe the improvement of the model on the noisy signal. The degree of improvement on each noise type is added in a separate column in the table. According to the table, the average of improvement on RMSE is  $41\times$  and the average of improvement on PPE is  $58\times$ .

An example of a reconstructed signal is presented in Figure 8, together with the noisy PPG and the reference PPG signal. As we can see in this figure, the noise is significantly reduced, and the peak values are adjusted accordingly, confirming that the image transformation successfully represents the noise in a visual format.

#### 3.1 Comparison

In this section we compare our model's efficiency with the state-of-the-art (Table 3). To minimize the difference between our experimental setup and the setups used in the related works we use the same measures. Such comparison for the state-of-the-art artifact detection and artifact removal algorithms has been made comprehensively in [32], where the algorithms are compared according to their input/output PPE and RMSE. We use [32] as the base of our comparison and we provide PPE and RMSE improvements of our method to display its efficiency with respect to both measures. It should be noted that it is not feasible to perform a close comparison between

Table 2. Results of the proposed method.

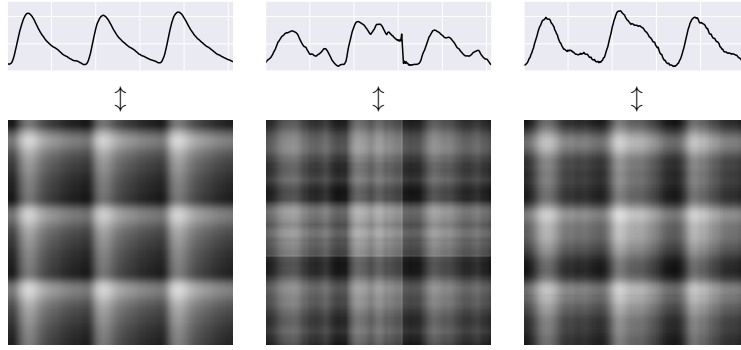
Noise Type	S/N (dB)	RMSE Gen. (BPM)	RMSE Nsy. (BPM)	RMSE Imprv.	PPE Gen. (BPM)	PPE Nsy. (BPM)	PPE Imprv.
Waving	20.04	0.213	41.76	196.07×	0.136	32.89	241.60×
Waving	11.30	2.43	55.30	22.75×	1.088	37.90	34.84×
3D Arm Movement	20.17	1.644	92.12	56.03×	0.772	44.03	57.06×
3D Arm Movement	13.12	1.688	65.99	39.10×	0.700	48.49	69.29×
Shaking Hands	21.66	1.556	61.89	39.78×	0.576	28.62	49.71×
Shaking Hands	14.96	4.203	84.31	20.06×	2.677	64.58	24.12×
Finger Tapping	22.99	1.758	63.43	36.07×	0.653	45.14	69.14×
Finger Tapping	13.99	3.008	21.76	7.235×	1.191	10.70	8.99×
Fist Open Close	25.11	1.648	35.74	21.69×	0.528	24.51	46.44×
Fist Open Close	16.69	2.151	51.28	23.84×	1.113	42.65	38.33×
Running Arm	20.14	2.056	22.93	11.16×	0.715	19.32	27.02×
Running Arm	13.98	3.807	77.73	20.42×	1.348	50.75	37.64×
Average	17.85	2.18	56.19	41.18×	0.958	37.465	58.68×

our model and the existing works, due to the differences in the datasets and the lack of a public dataset providing noisy and clean signals simultaneously.

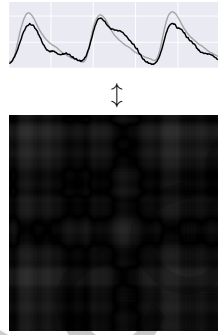
Table 3. The summary comparison of our result with the existing methods. MAE stands for Mean absolute error.

Paper	Method	Accelerometer	Before	Outcome
<b>Proposed method</b>	<b>CycleGAN</b>	<b>No</b>	<b>PPE 37.46 BPM</b> <b>RMSE 56.18 BPM</b>	<b>PPE 0.95 BPM</b> <b>RMSE 2.18 BPM</b>
Hanyu and Xiaohui [17]	Statistical Evaluation	No	PPE 8.1 BPM	PPE 7.85 BPM
Bashar et al. [12]	VFCDM	No	N/A	6.45% false positive
Lin and Ma [25]	DWT	No	PPE 13.97 BPM	PPE 6.87 BPM
Raghuram et al. [33]	CEMD LMS	Syn.	PPE 0.466 BPM	PPE 0.392 BPM
Hara et al. [18]	NLMS and RLS	Syn.	RMSE 28.26 BPM	RMSE 6.5 BPM
Tanweer et al. [37]	SVD and X-LMS	Yes	N/A	PPE 1.37 BPM
Wu et al. [39]	DC remover and RLS	Yes	N/A	STD 3.81
Baca et al. [11]	MAR and AT	Yes	N/A	MAE 2.26 BPM
Askari et al. [10]	SSA + MA Removal	No	N/A	RMSE 6.73 BPM

In comparison to non-accelerometer-based methods, our model significantly outperforms these models. The best performance observed in previous work is reported in [18] that improves the average RMSE from 28.26BPM to 6.5BPM (4.3× improvement). However, our model's improvement on average RMSE is from 56.18 to 2.18 (25.8× improvement). In most of the existing accelerometer-based methods, no value is provided for the degree of the input noise. Although the best reported PPE belongs to [33] with an outcome of 0.392BPM, the best improvement is achieved by [25] from 13.97BPM to 6.87BPM (2.03× improvement). However, our model's improvement on average PPE is from 37.46BPM to 0.95BPM (39.4× improvement).



(a) The clean reference signal of Figure 7 on the left, alongside its noisy signal in the middle, and the reconstructed clean signal on the right.



(b) A comparison of the reference signal and the reconstructed signal together with the residual image of the reconstruction is included in this figure. The brighter the pixels are the more difference the reconstructed image has in that pixel compared to the original image, meaning that the motion artifact is slightly apparent in the reconstructed signal.

Fig. 8. Signal reconstruction

### 3.2 Resource Usage

In this section, we provide detailed information about the resources being used. Previously, we claimed that our implemented model consumes a lower amount of power in comparison to the accelerometer-based models. We designed an experiment to measure the consumed power for our implemented model and accelerometer-based models to compare power consumption. First, we did 32Hz-sampling in a Raspberry Pi 4 device for five minutes by using a low-power accelerometer, ADXL343 [1]. Then, we measured the average consumed power of this task by using SmartPower2 5VDC Power Supply [3]. Secondly, we measured the average power consumption of our CycleGAN model in the test phase for five minutes with the same raspberry pi and power analyzer. In other words, we did these sub-tasks (1) trained the model, (2) tested our pre-trained model on the raspberry pi, and (3) monitored the power consumption. For embedded devices, the critical key is being low power in the training phase is not crucial; since training can be done on the cloud instead of the device itself. In table 4, you can find the results of this experiment. According to Table 4, our proposed method in average uses 45% less power compared to an accelerometer-based artifact removal method.

Table 4. Average power consumption of raspberry pi 4

	Idle	Accelerometer-based	<b>Proposed Method</b>
Power Consumption	2.23 W	3.76 W	<b>3.07 W</b>

To complete the analysis for resource usage, we also considered average time and memory consumption. We completed our proposed model's training and testing phase on a server, with a GPU of RTX 3080 Ti and a CPU of Xeon E5-2680 v2, and monitored the resources. In table 5, the information about resource usage is provided. Since our purpose is cleaning the PPG signal in real-time, low time consumption in the test phase is important. Based on the results, our model needs just 0.3 Sec to clean the PPG signal, which makes it a feasible solution for real-time PPG noise removal.

Table 5. Average time and memory consumption of the proposed method in training and testing phase

	Time	CPU Memory	GPU Memory
Train	909.786 Sec	488 MB	10828MB
Test	<b>0.398 Sec</b>	26 MB	1MB

#### 4 DISCUSSION

Noise reduction has been extensively studied in image processing, and the introduction of powerful models such as CycleGAN has shown promising results in terms of noise reduction in images. Inspired by this fact, we proposed a signal to image transformation that visualizes signal noises in the form of image noise. To the best of our knowledge, this is the first use of CycleGAN for bio-signal noise reduction which eliminates the need for an accelerometer to be embedded into wearable devices, which in turn helps to reduce the power consumption and cost of these devices.

It should be noted that despite the significant benefits of our proposed method in removing noise in different situations, it may not be effective in all possible scenarios. Clearly, the intensity of noise applied to the signals, and the variations of the noise, also called noise categories, are controlled for the purpose of this study. In other words, if the source of the generated motion artifact is changed in a way that the range in heart rate variations is observable in compared with existing activities in this work, this method may not be applicable. Although it will improve the error, it does not guarantee a reasonable upper bound. However, the same limitations also exist in the related works.

#### 5 CONCLUSIONS

In this paper, we presented an image processing approach to the problem of noise removal from PPG signals where the noise is selected from a set of noise categories that simulate the daily routine of a person. This method does not require an accelerometer on the sensor, therefore, it can be applied to other variations of physiological signals, such as ECG, to reduce the power usage of the measuring device and improve its efficiency. In this work, the novel use of CycleGAN as an image transformer is leveraged to transform such physiological signals. On average, the reconstructed PPG performed using our proposed method offers 41× improvement on RMSE and 58× improvement on PPE, outperforming the state-of-the-art by a factor of 9.5.

## REFERENCES

- [1] [n.d.]. Analog Devices | ADXL343. <https://www.analog.com/media/en/technical-documentation/data-sheets/adxl343.pdf>.
- [2] [n.d.]. Empatica | Medical devices, AI and algorithms for remote patient monitoring. <https://www.empatica.com/>. Accessed: 2021-05-24.
- [3] [n.d.]. SmartPower2 5VDC Power Supply. <https://ameridroid.com/products/smartpower2-5vdc-power-supply>.
- [4] John Allen. 2007. Photoplethysmography and its application in clinical physiological measurement. *Physiological measurement* 28, 3 (2007), R1.
- [5] Seyed Amir Hossein Aqajari, Rui Cao, Emad Kasaeyan Naeini, Michael-David Calderon, Kai Zheng, Nikil Dutt, Pasi Liljeberg, Sanna Salanterä, Ariana M Nelson, and Amir M Rahmani. 2021. Pain assessment tool with electrodermal activity for postoperative patients: Method validation study. *JMIR mHealth and uHealth* 9, 5 (2021), e25258.
- [6] Seyed Amir Hossein Aqajari, Rui Cao, Amir Hosein Afandizadeh Zargari, and Amir M Rahmani. 2021. An End-to-End and Accurate PPG-based Respiratory Rate Estimation Approach Using Cycle Generative Adversarial Networks. *arXiv preprint arXiv:2105.00594* (2021).
- [7] Seyed Amir Hossein Aqajari, Emad Kasaeyan Naeini, Milad Asgari Mehrabadi, Sina Labbaf, Nikil Dutt, and Amir M Rahmani. 2021. pyEDA: An Open-Source Python Toolkit for Pre-processing and Feature Extraction of Electrodermal Activity. *Procedia Computer Science* 184 (2021), 99–106.
- [8] Milad Asgari Mehrabadi, Seyed Amir Hossein Aqajari, Amir Hosein Afandizadeh Zargari, Nikil Dutt, and Amir M Rahmani. 2022. Novel Blood Pressure Waveform Reconstruction from Photoplethysmography using Cycle Generative Adversarial Networks. *arXiv e-prints* (2022), arXiv-2201.
- [9] Marzieh Ashrafiamiri, Sai Manoj Pudukotai Dinakarrao, Amir Hosein Afandizadeh Zargari, Minjun Seo, Fadi Kurdahi, and Houman Homayoun. 2020. R2AD: Randomization and Reconstructor-based Adversarial Defense on Deep Neural Network. In *Proceedings of the 2020 ACM/IEEE Workshop on Machine Learning for CAD*. 21–26.
- [10] Mohammad Reza Askari, Mudassir Rashid, Mert Sevil, Iman Hajizadeh, Rachel Brandt, Sediqeh Samadi, and Ali Cinar. 2019. Artifact removal from data generated by nonlinear systems: Heart rate estimation from blood volume pulse signal. *Industrial & Engineering Chemistry Research* 59, 6 (2019), 2318–2327.
- [11] Alessandro Baca, Giorgio Biagetti, Marta Camilletti, Paolo Crippa, Laura Falaschetti, Simone Orcioni, Luca Rossini, Dario Tonelli, and Claudio Turchetti. 2015. CARMA: a robust motion artifact reduction algorithm for heart rate monitoring from PPG signals. In *2015 23rd European signal processing conference (EUSIPCO)*. IEEE, 2646–2650.
- [12] Syed Khairul Bashar, Dong Han, Apurv Soni, David D McManus, and Ki H Chon. 2018. Developing a novel noise artifact detection algorithm for smartphone PPG signals: Preliminary results. In *2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 79–82.
- [13] Rui Cao, Seyed Amir Hossein Aqajari, Emad Kasaeyan Naeini, and Amir M Rahmani. 2021. Objective pain assessment using wrist-based ppg signals: A respiratory rate based method. In *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 1164–1167.
- [14] Jingwen Chen, Jiawei Chen, Hongyang Chao, and Ming Yang. 2018. Image blind denoising with generative adversarial network based noise modeling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3155–3164.
- [15] Farzam Ebrahimnejad and James R Lee. 2021. Multiscale entropic regularization for MTS on general metric spaces. *arXiv preprint arXiv:2111.10908* (2021).
- [16] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).
- [17] Shao Hanyu and Chen Xiaohui. 2017. Motion artifact detection and reduction in PPG signals based on statistics analysis. In *2017 29th Chinese control and decision conference (CCDC)*. IEEE, 3114–3119.
- [18] Shinsuke Hara, Takunori Shimazaki, Hiroyuki Okuhata, Hajime Nakamura, Takashi Kawabata, Kai Cai, and Tomohito Takubo. 2017. Parameter optimization of motion artifact canceling PPG-based heart rate sensor by means of cross validation. In *2017 11th international symposium on medical information and communication technology (ISMICT)*. IEEE, 73–76.
- [19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
- [20] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1125–1134.
- [21] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*. Springer, 694–711.
- [22] Kushal Joshi, Alireza Javani, Joshua Park, Vanessa Velasco, Binzhi Xu, Olga Razorenova, and Rahim Esfandypour. 2020. A Machine Learning-Assisted Nanoparticle-Printed Biochip for Real-Time Single Cancer Cell Analysis. *Advanced Biosystems* 4, 11 (2020), 2000160.
- [23] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4681–4690.

- [24] Chuan Li and Michael Wand. 2016. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *European conference on computer vision*. Springer, 702–716.
- [25] Wei-Jheng Lin and Hsi-Pin Ma. 2016. A physiological information extraction method based on wearable PPG sensors with motion artifact removal. In *2016 IEEE international conference on communications (ICC)*. IEEE, 1–6.
- [26] Stefan CB Mannsfeld, Benjamin CK Tee, Randall M Stoltenberg, Christopher V Chen, Soumendra Barman, Beinn VO Muir, Anatoliy N Sokolov, Colin Reese, and Zhenan Bao. 2010. Highly sensitive flexible pressure sensors with microstructured rubber dielectric layers. *Nature materials* 9, 10 (2010), 859–864.
- [27] Ahmadreza Moradipari, Mahnoosh Alizadeh, and Christos Thrampoulidis. 2020. Linear thompson sampling under unknown linear constraints. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 3392–3396.
- [28] Ahmadreza Moradipari, Sanae Amani, Mahnoosh Alizadeh, and Christos Thrampoulidis. 2021. Safe linear thompson sampling with side information. *IEEE Transactions on Signal Processing* 69 (2021), 3755–3767.
- [29] Ahmadreza Moradipari, Christos Thrampoulidis, and Mahnoosh Alizadeh. 2020. Stage-wise conservative linear bandits. *Advances in neural information processing systems* 33 (2020), 11191–11201.
- [30] Alireza Nikzamir and Filippo Capolino. 2022. Highly Sensitive Coupled Oscillator Based on an Exceptional Point of Degeneracy and Nonlinearity. *arXiv preprint arXiv:2206.04031* (2022).
- [31] Marco AF Pimentel, Alistair EW Johnson, Peter H Charlton, Drew Birrenkott, Peter J Watkinson, Lionel Tarassenko, and David A Clifton. 2016. Toward a robust estimation of respiratory rate from pulse oximeters. *IEEE Transactions on Biomedical Engineering* 64, 8 (2016), 1914–1923.
- [32] David Pollreisz and Nima TaheriNejad. 2019. Detection and removal of motion artifacts in PPG signals. *Mobile Networks and Applications* (2019), 1–11.
- [33] M Raghuram, Kosaraju Sivani, and K Ashoka Reddy. 2016. Use of complex EMD generated noise reference for adaptive reduction of motion artifacts from PPG signals. In *2016 international conference on electrical, electronics, and optimization techniques (ICEEOT)*. IEEE, 1816–1820.
- [34] Monalisa Singha Roy, Rajarshi Gupta, Jayanta K Chandra, Kaushik Das Sharma, and Arunansu Talukdar. 2018. Improving photoplethysmographic measurements under motion artifacts using artificial neural network for personal healthcare. *IEEE Transactions on Instrumentation and Measurement* 67, 12 (2018), 2820–2829.
- [35] Mert Sevil, Mudassir Rashid, Iman Hajizadeh, Mohammad Reza Askari, Nicole Hobbs, Rachel Brandt, Minsun Park, Laurie Quinn, and Ali Cinar. 2021. Discrimination of simultaneous psychological and physical stressors using wristband biosignals. *Computer Methods and Programs in Biomedicine* 199 (2021), 105898.
- [36] Sina Shahsavari, Pulak Sarangi, and Piya Pal. 2021. KR-LISTA: Re-Thinking Unrolling for Covariance-Driven Sparse Inverse Problems. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 1403–1408.
- [37] Khawaja Taimoor Tanweer, Syed Rafay Hasan, and Awais Mehmood Kamboh. 2017. Motion artifact reduction from PPG signals during intense exercise using filtered X-LMS. In *2017 IEEE international symposium on circuits and systems (ISCAS)*. IEEE, 1–4.
- [38] Linh Duy Tran, Son Minh Nguyen, and Masayuki Arai. 2020. GAN-based Noise Model for Denoising Real Images. In *Proceedings of the Asian Conference on Computer Vision*.
- [39] Chih-Chin Wu, I-Wei Chen, and Wai-Chi Fang. 2017. An implementation of motion artifacts elimination for PPG signal processing based on recursive least squares adaptive filter. In *2017 IEEE biomedical circuits and systems conference (BioCAS)*. IEEE, 1–4.
- [40] Rozhin Yasaei, Luke Chen, Shih-Yuan Yu, and Mohammad Abdullah Al Faruque. 2022. Hardware Trojan Detection using Graph Neural Networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* (2022).
- [41] Rozhin Yasaei, Felix Hernandez, and Mohammad Abdullah Al Faruque. 2020. IoT-CAD: context-aware adaptive anomaly detection in IoT systems through sensor association. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*. IEEE, 1–9.
- [42] Amir Hosein Afandizadeh Zargari, Manik Dautta, Marzieh AshrafiAmiri, Minjun Seo, Peter Tseng, and Fadi Kurdahi. 2020. NEWERTRACK: ML-Based Accurate Tracking of In-Mouth Nutrient Sensors Position Using Spectrum-Wide Information. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 39, 11 (2020), 3833–3841.
- [43] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*. 2223–2232.