



PBVI for Optimal Photoplethysmography Noise Filter Selection Using Human Activity Recognition Observations for Improved Heart Rate Estimation on Multi-Sensor Systems

Jacob J. Sindorf

School of Manufacturing Systems and Networks,
Arizona State University,
Mesa, AZ 85212

Sangram Redkar

School of Manufacturing Systems and Networks,
Arizona State University,
Mesa, AZ 85212

This work details the partially observable markov decision process (POMDP) and the point-based value iteration (PBVI) algorithms for use in multisensor systems, specifically, a sensor system capable of heart rate (HR) estimation through wearable photoplethysmography (PPG) and accelerometer signals. PPG sensors are highly susceptible to motion artifact (MA); however, current methods focus more on overall MA filters, rather than action specific filtering. An end-to-end embedded human activity recognition (HAR) System is developed to represent the observation uncertainty, and two action specific PPG MA reducing filters are proposed as actions. PBVI allows optimal action decision-making based on an uncertain observation, effectively balancing correct action choice and sensor system cost. Two central systems are proposed to accompany these algorithms, one for unlimited observation access and one for limited observation access. Through simulation, it can be shown that the limited observation system performs optimally when sensor cost is negligible, while limited observation access performs optimally when a negative reward for sensor use is considered. The final general framework for POMDP and PBVI was applied to a specific HR estimation example. This work can be expanded on and used as a basis for future work on similar multisensor system. [DOI: 10.1115/1.4065219]

Keywords: medical instrumentation, minimally invasive devices, sensors/actuators

1 Introduction

Variation of the body's blood volume due to the cardiac cycle is a basis for the PPG signal. PPG signals are derived through extracutaneous optical sensors that irradiate light which is absorbed or unabsorbed by the body. The optical sensor then receives the unabsorbed light through either transmission or reflection, forming a PPG signal. As mentioned, PPG signals contain information about the body's cardiac cycle, the change in blood volume directly alter the light absorption. Thus, the PPG waveform is synchronous with each heartbeat [1,2].

As PPG signals relate to the cardiac cycle, they are used in many medical diagnostics and testing. A critical interpretation of the PPG signal is HR in beats per minute (BPM). HR information remains an essential tool at home and in clinic diagnostics as it can be derived from an inexpensive and noninvasive sensor system. To estimate HR from PPG, the most common method would be to use the time

between peaks and the signal's sampling frequency, f_s . The following equation details HR estimation

$$\text{BPM} = \frac{f_s}{\text{Peak}(i) - \text{Peak}(i-1)} \times 60 \quad (1)$$

where f_s is the sampling frequency the signal was obtained at, Peak (i) refers to the current maximum peak of the waveform, and Peak (i-1) refers to the previous maximum waveform peak. This method, shown in Eq. (1), provides a simple and direct way to calculate HR from PPG, given a clean, uncorrupted waveform is used; However, PPG is obtained via an externally worn sensor, usually on the wrist or fingertips. Sensor location causes the sensor signal to be highly susceptible to MA, which alters the PPG signal dynamics, forcing inaccurate HR estimation and rendering Eq. (1) useless. In activities that do not require the user to be stationary, such as walking or running, HR estimation can be challenging. The following Fig. 1 displays a PPG signal from a finger-worn device in four activities: Sitting, standing, walking, and running. Along with the PPG signal, triaxial accelerometer signals are also shown to visualize the motion near the PPG collection site.

Manuscript received April 10, 2023; final manuscript received March 25, 2024; published online April 15, 2024. Assoc. Editor: Jun Liao.

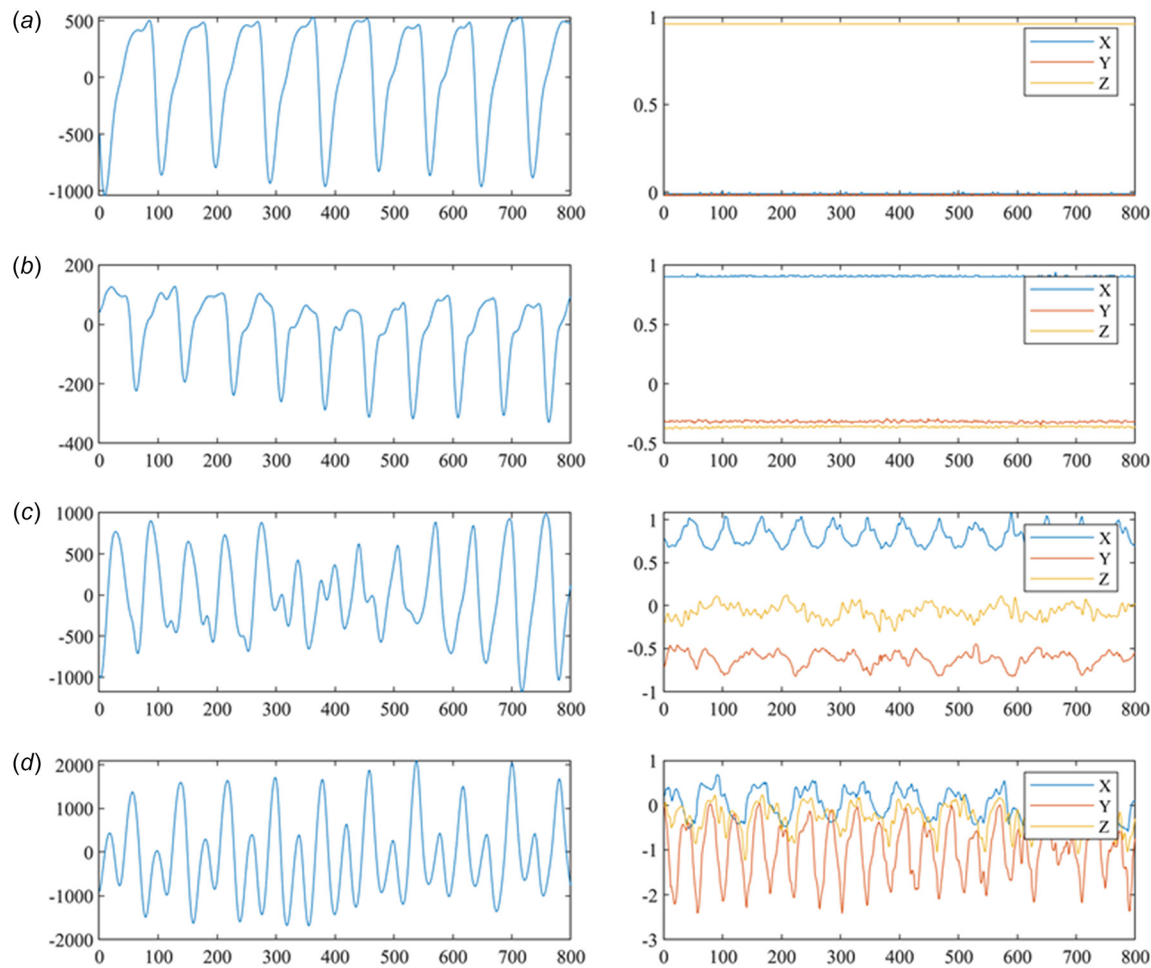


Fig. 1 PPG and accelerometer data per activity. PPG (left) and triaxial accelerometer data (right) for four activities: (a) sitting, (b) standing, (c) walking, and (d) running.

As seen in Fig. 1, the degree of motion strongly alters the PPG signal obscuring the overall waveform. To produce accurate HR estimations on high-motion activities, it is imperative to create robust filters to reduce the effects of MA on PPG waveforms. There have been multiple effective approaches to reducing or removing MA from PPG. Many techniques revolve around some variation of the least mean squares (LMS) filter [3,4] or the application of independent component analysis [5], or singular value decomposition in conjunction with LMS [6,7]. Other techniques to reduce MA from PPG include using the frequency domain [8], wavelets, recursive least squares, or the Hankel matrix [9]. These methods, although successful, do not optimize MA reduction per activity, ignoring vital differences in the activities and their effect on the magnitude of MA on the signal. Targeted filters based on the user's current activity, i.e., HAR, would reduce the risk of over-filtering the signal by applying activity specific filters.

Knowing the state of the user, such as activity from HAR, would then require the system to make optimal decisions, or actions, by applying the appropriate filter. In parallel, wearable sensor systems or wireless body area networks typically have one central power unit making energy savings on top of optimal decision making critical. This is where success with dynamic programming (DP) has been found. DP can be used to find accurate, and low-energy ways to use multisensor systems [10,11]. Another technique called Markov decision process (MDP), which uses probabilities to determine optimal actions, has also found success in energy savings in multisensor systems, with one particular case using a PPG sensor [12]. The results showed promising energy savings using an MDP method indicating the use cases of DP and MDP in multisensor systems for HAR and HR estimation.

This work proposes a method to apply frequency-based noise filters to reduce the effects of MA based on the observed user's activity. This method chooses optimal actions of PPG filter selection based on the user's current activity to allow long-term, accurate HR signal readings. To solve this, a POMDP and a point-based value iteration (PBVI) framework are proposed. Usage of the POMDP and PBVI algorithms are supplemented with a practical example. The practical example includes an end-to-end embedded HAR device and two specific MA-reducing filters for PPG.

2 Methods

To optimally select the correct action, POMDP, and PBVI are used and have been well described by Pineau et al. [13]. A POMDP closely follows a MDP except the agent can only perceive observations that can be used to convey incomplete information about the state. Common terminology to describe a POMDP includes states, S , actions, U , and observations, Y . A state transition probability, $P_{tr} = \mathbb{P}(S_{k+1} = j | S_k = i, U_k = u)$, an observation probability distribution, $P_o = \mathbb{P}(Y_k = y | S_k = i, U_k = u)$, and a reward function, $\mathbb{E} \left[\sum_{k=0}^{N-1} r \right]$.

Application of PBVI curbs the curse of dimensionality of POMDP. To start, a set of beliefs can be defined as the probability distribution of the current state and history. The belief can be defined as a sufficient statistic to predict the future. However, the belief space, $\mathbb{B}$ can be large, bringing the need for a term, $\mathcal{A}$, to represent a suitable finite-dimensional set of vectors of α . The value function update can be implemented as a sequence of operations on the set of vectors. However, this still contains a large number of vectors. Thus,

a relevant subset of vectors, $\hat{\mathcal{A}}$, can be used on a smaller dimensional subset of belief points, $\mathcal{B}$. PBVI starts with a small initial set of belief points to start a set of backup operations. Belief points grow, and backup operations continue until an approximate solution is reached. The equations and methods by Pineau et al. [13] are explained further in the application of this work.

To correctly fit the system model into a POMDP framework, the system states, actions, rewards, transitions, and observation probabilities must be described. This work compares two separate systems, denoted A: Unlimited Observation Access and B: Limited Observation Access.

Both systems share the same state space and transition probabilities. Transition Probabilities can be described as $P_{tr} = \mathbb{P}(S_{k+1} = j | S_k = i, U_k = u)$, where $S_{k+1} = j$ represents the next state, $S_k = i$ represents the current state, and $U_k = u$ represents the current action. The probabilities and states are modeled after the ones derived by Thatte et al. [10] and displayed in the Markov Chain from Zois et al. [11]. States have been simplified to $\mathcal{S} \in \{1, 2, 3, 4\}$, corresponding to sit, stand, walk, and run, respectively. The probabilities are action independent and can be simplified into the following transition probability matrix:

$$P_{tr} = \mathbb{P}(S_{k+1} = j | S_k = i) = P_{ij} = \begin{bmatrix} .6 & .1 & .3 & 0 \\ .2 & .4 & .3 & .1 \\ .4 & 0 & .3 & .3 \\ 0 & .1 & .6 & .3 \end{bmatrix}$$

Generally, the actions can be described as the PPG filter selection for both systems. The system may choose one of two filters to apply to the PPG signal based on the user's current state, using no filter for the low motion, quasi-static states of sit and stand and a unique noise-reducing filter for walk and run. As mentioned, the system follows observations of the state, meaning the actual state of the user is not always correctly observed. To accompany the POMDP and PBVI framework, this work provides practical applications for both the observation estimation and the application of the PPG noise filter. First, an end-to-end HAR-embedded machine learning system is created to represent the uncertainty of observations. Next, two PPG filters optimized for walk and run states were developed and tested. Then using the practical examples of observation uncertainty and PPG filters, system A and system B are described in detail for use in the POMDP and PBVI framework.

2.1 Application: End-to-End Human Activity Recognition.

An end-to-end human activity recognition (HAR) embedded system was created to provide a real basis for the observation's uncertainty in representing the state. The observation is based on the estimated state or observation the system receives, as it does not have direct access to the actual states sit, stand, walk, and run. Without direct access to the actual state, or in this case, activity, the system must estimate the state as an observation, which inherently comes with some uncertainty and inaccuracy. A HAR-embedded system can be used to model the uncertainty and estimation accuracy of the true state with an observation. Thus, the final validation accuracy the HAR system achieves represents how accurate observation is when estimating state. Embedded machine learning was used to create the HAR model, which requires data collection, data processing, model training, and model deployment, which can all be performed through Edge Impulse [14].

An Arduino Nano 33 BLE sense was used to collect data as it contains a built-in inertial measurement unit (IMU), the LSM9DS1, and can run embedded machine learning. The IMU provides triaxial accelerometer, triaxial gyroscope, and triaxial magnetometer data giving nine total input data sources. Based on the states, sit, stand, walk, and run, IMU data needed to be collected performing each activity in a wrist-like location. In detail: Sitting, with arms on a table or in their lap with small to no arm motion, standing, with arms at the side with small body motions like swaying, walking, at

different paces, from 1 to 4 mph, and running, at different paces, from 4.5 to 7 mph.

The Arduino can then be connected to Edge Impulse, where the IMU can be directly read, and the data can be labeled for use in a classification problem. IMU data can be collected for each activity in 5-second intervals at 100Hz. In total, 25 min of data can be collected, with 6.25 min of data per activity. An 80/20 train test split can be used on the data, with each 5-second sample being windowed with a window size of 2 s and a window increase of 200 ms.

The following spectral settings were chosen through Edge Impulse's built-in preprocessing blocks. For each of the nine inputs, the axis was scaled by 2, a lowpass filter with a 5 Hz cutoff and a 4th order Butterworth were applied, and 64 FFT points were considered. These settings most optimally separated the features as shown in Fig. 2. After feature extraction, 54 final features were selected and fed through a simple neural network. The network consisted of an input layer, two hidden layers of 20 and 10 neurons, respectively, and an output layer with a dimension of four to represent the four states. The model ran for 300 iterations with a learning rate of 0.0005 with a final accuracy of 99.7% and a final categorical cross-entropy loss of 0.01.

With a fully trained model, a validation set consisting of 20% of the training data can be run on the system to validate the performance. After validation, a final validation accuracy of 96.25% was achieved. Most often, walking was mistaken for running (3.8%) while standing was mistaken for walking (6.3%). The final system can generate observations that correctly identify the state approximately 96% of the time, giving a real basis for observation space generation. This uncertainty can be represented by the value ϵ where $\epsilon = 1 - \text{validation accuracy}$ yielding $\epsilon = 0.04$.

The HAR system can be fully embedded by deploying the trained model to Arduino and adding a power source to the Arduino and securing it to the wrist. Through Edge Impulse, the trained model can then be packaged as an Arduino library file and exported to the Arduino IDE. The IMU can be read to build an input tensor of IMU data matching the 2-second window described. The input tensor can then be fed to the model to output continuous HAR through Bluetooth or serial connection to a different device.

2.2 Application: Motion Artifact Filtering of Photoplethysmography for Accurate Heart Rate Estimation. As discussed, MA on PPG signals obscure the underlying waveform and force inaccurate estimation of HR. Low motion activities such as sitting and standing allow for clean PPG signals as there would be negligible amounts of MA corrupted signal. Thus, Eq. (1) can be used directly on any portion of data that includes two peaks, and no additional filtering would be required to get direct HR values. This method can be improved by taking the average peak-to-peak across a window of data, as there may be some rare instances of an inaccurate waveform.

Higher motion activities such as walking and running are where unique MA removal filters can be applied. As shown in Fig. 1, the degree of MA on the PPG signal and the amplitude of the

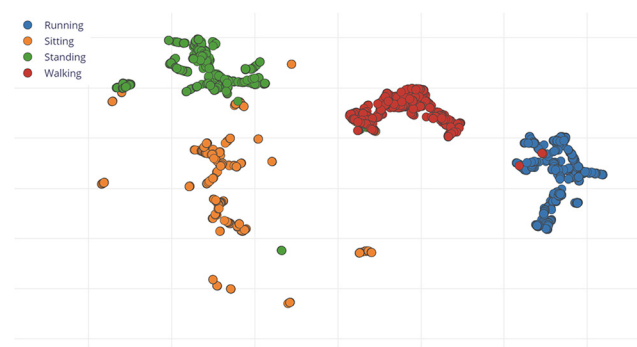


Fig. 2 Extracted activity features visualized with edge impulse

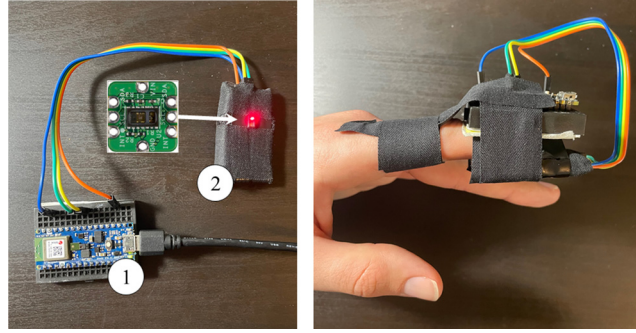


Fig. 3 Device hardware used to collect PPG and accelerometer data and for performing embedded HR estimation. On the left are the hardware components, where (1) is the Arduino Nano 33 BLE Sense and (2) is the MAXREFDES117 PPG evaluator with a foam stabilizer. The right depicts the device being worn on the left index finger.

accelerometer differ between walking and running. To develop and test optimal noise filters, example data, including PPG and triaxial accelerometer, were collected on the two activities, walk and run. The data can be collected on the finger-worn device shown in Fig. 3. To collect PPG signals, a MAXREFDES117 PPG evaluator was used, while sensor processing and accelerometer were collected from an Arduino Nano 33 BLE Sense. Both PPG and accelerometer were collected around 104 Hz.

2.2.1 Photoplethysmography Filter Methods. Both final filters, named Filter 1: Walking and Filter 2: Running in this work, were created using two methods, sliding window and frequency cutting. The sliding window method begins by gathering a window of data across the PPG and triaxial accelerometer data. Given the 104 Hz sampling rate for both sensors, a full window of data was selected to contain 1000 samples or approximately 10 s of data. The window then slides 100 samples, or around 1 s getting 100 new samples and removing 100 old samples.

Frequency cutting requires two steps, frequency range reduction, and accelerometer correlation reduction, which are applied to each PPG data window of 1000 samples. The first step, frequency range reduction, reduces frequencies from specified ranges. The 0–0.5 Hz range is the first to be reduced, which corresponds to around 0 to 30 BPM, using a parameter named R_{Low} by applying $\frac{1}{R_{Low}}$ to the PPG FFT amplitude. Next, the 4 Hz to Nyquist Frequency, which equates to 52 Hz or half the sampling rate, is reduced by a parameter named R_{High} by applying $\frac{1}{R_{High}}$ to the PPG FFT amplitude. The final frequency reduction parameter R_{Mid} is applied only to the Running filter as it reduces frequencies from .5 to 1.33 Hz or the 30 to 80 BPM range.

The second step in frequency cutting is accelerometer correlation reduction. To start, the FFT of the triaxial accelerometer, including X, Y, and Z, must be taken individually. The resulting FFTs can be added to create a total accelerometer FFT. The absolute total accelerometer FFT and absolute PPG FFT peaks are then compared. If the peak from the accelerometer matches the peak from the PPG, then a parameter named R_{ACC} or R_{ACCR} is applied as $\frac{1}{R_{ACC}}$ or $\frac{1}{R_{ACCR}}$ to the PPG FFT amplitude to both the matching peak, and the values just before and after it. $\frac{1}{R_{ACC}}$ is applied when the accelerometer peak is greater than the PPG peak, while $\frac{1}{R_{ACCR}}$ is applied when the PPG peak is greater than the accelerometer. The most significant difference between walking and running here is that walking also considers peaks that are off by one when reducing the PPG FFT, while running only considers peaks that exactly match.

The following, Fig. 4, display the frequency cuttings effect on the PPG's FFT for walking and running, respectively. From Fig. 4, it is possible to see the frequency range reduction and the accelerometer correlation reduction on a snapshot of data. The differences between the walking and running filters can also be seen. From Fig. 4, the accelerometer and PPG peaks that are off by 1 are considered and reduced while they are not in the running filter. From Fig. 4, the R_{Mid} frequency range reduction parameter is used but is not used on walking.

After taking a window of data and performing frequency cutting, the HR can be estimated from the resulting reconstructed PPG signal. This is performed by taking the inverse FFT of the data

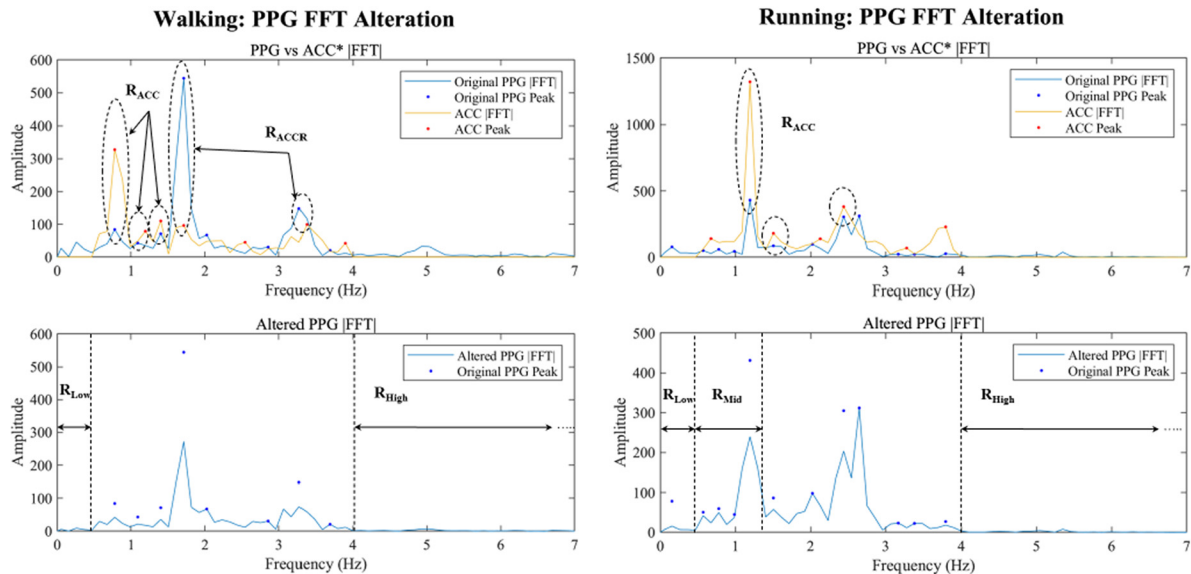


Fig. 4 Walking (left) and running (right) frequency cutting method example on the PPG signal from set 1. The top graph compares the original PPG absolute FFT and the accelerometer absolute FFT. The matching peaks are circled, and the reduction parameters, R_{ACC} and R_{ACCR} are shown. R_{ACC} refers to the accelerometer amplitude being greater than the PPG, while R_{ACCR} refers to the PPG amplitude being greater than the accelerometer. It can be seen that some accelerometer and PPG peaks are offset and still circled as matching. This was intentionally done to the walking filter to improve performance. Note that the accelerometer, called ACC, has had the 0–0.5 Hz and 4 Hz plus range set to zero, the bottom graph shows the altered absolute PPG FFT. Shown are the ranges for the R values, R_{Low} , R_{Mid} , R_{High} . Both frequency range reduction and accelerometer correlation reduction have been applied.

window. To estimate HR, Eq. (1) can be used by taking the average peak-to-peak distance across all waveforms contained in the window. This averaged value can then be used with the sampling rate of 104 Hz to produce an estimated HR for that window of data. The sliding window then shifts 100 samples, and the process is repeated until five shifts have occurred and five estimations have been created. These five estimations are then averaged and output as the final estimated HR for the approximately 5 s it took to gather the new 500 data points. A final estimation technique is included, which allows the filters to be more robust to large changes in HR estimates. The filter checks the previous HR estimation, which was the average of five sliding windows, and compares it to the current estimate. If the estimations have a difference greater than 10 BPM, the new estimation is taken as the midpoint between estimations, as shown in Eq. (2), where i represents the current estimation, and $i - 1$ represents the previous estimation

$$HR_{est}(i) = \frac{HR_{est}(i) + HR_{est}(i-1)}{2} \quad (2)$$

The walking and running filters have some differences, as mentioned previously; however, the main difference between the filters lies with the system parameters. Along with the reduction parameters, R , the peak prominence when estimating HR, P_{prom} , must also be considered. Peak prominence determines the magnitude of peaks to detect when performing peak-to-peak calculations. These system parameters can be tuned to optimally estimate HR for a particular activity. They can also be tuned to be subject-specific, so the filter can ensure optimal HR estimation per subject.

2.2.2 Photoplethysmography Filter Outcomes. To determine the performance of the filters on walking and running, both are compared to just the sliding window. These are then validated by the ground truth HR value from the NONIN WristOx2 [15,16] through mean absolute error (MAE) shown in Eq. (3) where N is the total number of estimations, n is the current estimation, BPM_{est} is the estimated HR, and BPM_{NONIN} is the true HR from NONIN.

$$MAE = \frac{1}{N} \sum_{n=1}^N |BPM_{est}(n) - BPM_{NONIN}(n)| \quad (3)$$

Two example sets were used to test the filter's ability to be tuned and optimized not only per activity but per set. Both sets performed an initial two minutes of the specified activity, then after two minutes, around one and a half minutes of sensor data and ground truth HR values were recorded. Given the sampling rate of around 104 Hz, ground truth HR was recorded every five seconds to match the sliding window averages for comparison. The device from Fig. 3 and the NONIN are index finger based.

The following table, Table 1, displays the exact parameter values used to optimize the filters for each activity on the data for set 1, while Table 2 displays parameters for set 2.

Filter 1: Walking, using the parameters from Tables 1 and 2 achieved an MAE of 4.31 and 5.48 BPM for sets 1 and 2,

Table 1 Filter parameter values used for the walking and running filters on set 1

Set 1 filter parameter comparison		
Parameter	Walking	Running
R_{Low}	5	5
R_{Mid}	1	1.2
R_{High}	5	5
R_{ACC}	2	1.5
R_{ACC}	2	1
R_{Prom}	0.9	0.2

Table 2 Filter parameter values used for the walking and running filters on set 2

Set 2 filter parameter comparison		
Parameter	Walking	Running
R_{Low}	5	5
R_{Mid}	1	1.2
R_{High}	5	5
R_{ACC}	5	1.5
R_{ACC}	5	1
R_{Prom}	0.5	0.7

respectively. Using just the sliding window achieved an MAE of 14.52 and 15.21 BPM showing a 10 BPM performance increase with the use of the final filter, named Filter 1: Walking.

Filter 2: Running, using the parameters from Tables 1 and 2, achieved an MAE of 3.98 and 3.06 BPM for sets 1 and 2, respectively. Using just the sliding window achieved an MAE of 8.43 and 7.99 BPM showing a 5 BPM performance increase.

Both Filters improved the HR estimation outcomes for the provided data; however, an MAE of around 5 BPM was achieved for both activities. This MAE value means that, on average, the estimation is incorrect by around 5 BPM from the ground truth. This means that although the filters do improve performance, they still come with a small estimation error when used.

The method of windowing, frequency cutting, and midpoint estimation reduction comes with specific parameters that can be tuned and optimized to fit the data best, motivating selective filtering. As shown with walking and running activity data, tuning the filter parameters to be activity-specific improves the HR estimation. The differences can be pointed toward the motion represented by the accelerometer, the difference in HR value for each activity, and even the difference in the user. Due to this, the frequency information would be different and require different filtering magnitudes to fit the activity best. From this, two activity-specific MA-reducing filters, named Filter 1: Walking and Filter 2: Running, were created to increase HR estimation accuracy.

With a basis for observation uncertainty from a HAR system and two specific noise-reducing filters for PPG described, differences in System A and System B can be described in detail in the subsequent sections.

2.3 System A: Unlimited Observation Access. System A observes the accelerometer readings at every step to make decisions meaning it has unlimited access to the observation space. It has three possible actions from an action space, $U \in \{1, 2, 3\}$, no filter used, filter 1: Walking used, and filter 2: Running used, respectively. As the states are not directly observable, it requires an observation space denoted as $Y \in \{1, 2, 3, 4\}$ to represent the interpreted observation of the state. As mentioned, the algorithm to correctly determine the state may not always be accurate, which is why an observation probability is used. The observation probability, $\mathbb{P}(Y_k = y | S_{k+1} = j, S_k = i, U_k = u)$, does not depend on action or next state and can simplify to:

$$P_o = \mathbb{P}(Y_k = y | S_k = i) = \begin{cases} \epsilon/3 \forall y \neq i \\ (1 - \epsilon) \forall y = i \end{cases}$$

Here, an $\epsilon = 0.04$ is used and was derived from the HAR application described. This again represents that 96% of the time, the observation can correctly estimate the state.

Lastly, the reward is denoted with $r(i, u)$, where i is the current state, and u is the current action. The reward structure for system A requires that the HAR system be kept on for continuous observation access, meaning every reward has a cost due to the HAR system's energy, called e_{HAR} . The correct choice of action yields a reward of R , and incorrect choice results in no reward. Actions 2 and 3 also turn

on the filter and gain a small cost due to the filter, labeled e_{filt} . The filter cost can be accredited to filter accuracy. Although the filters are robust, there may be rare instances where the signal is too corrupted for the filter to work correctly. This is also shown as the MAE results per filter were not perfectly zero, meaning each filter contains some degree of inaccuracy. The reward structure can be organized as follows:

$$r(i, u = 1) = \begin{cases} R - e_{\text{HAR}} & i = 1, 2 \\ 0 - e_{\text{HAR}} & i = 3, 4 \end{cases}$$

$$r(i, u = 2, 3) = \begin{cases} R - e_{\text{filt}} - e_{\text{HAR}} & i = u \\ 0 - e_{\text{filt}} - e_{\text{HAR}} & i \neq u \end{cases}$$

2.4 System B: Limited Observation Access. As stated, system B shares the same states and transition probabilities as system A; however, it has one main difference. That the system does not always have access to observations, which allows for reward comparison between the two models. The system now has an extra action in the action space, $U \in \{0, 1, 2, 3\}$, where action 0 activates the HAR system and allows access to the observation signals. This, in turn, expands the observation space to $Y \in \{0, 1, 2, 3, 4\}$, where the state estimation of 1 to 4 can only be accessed if action 0 is chosen, else it receives an observation value of 0. This is further shown in the observation model, which is now action dependent

$$P_o = \mathbb{P}(Y_k = y | S_k = i, U_k = 1, 2, 3) = \begin{cases} 1 & y = 0 \\ 0 & y \neq 0 \end{cases}$$

$$P_o = \mathbb{P}(Y_k = y | S_k = i, U_k = 0) = \begin{cases} \epsilon/3 & \forall y \neq i \\ (1 - \epsilon) & \forall y = i \\ 0 & y = 0 \end{cases}$$

System B also has a different reward structure, where only action 0 can access the accelerometer, as follows:

$$r(i, u = 0) = -e_{\text{HAR}} \forall i$$

$$r(i, u = 1) = \begin{cases} R & i = 1, 2 \\ 0 & i = 3, 4 \end{cases}$$

$$r(i, u = 2, 3) = \begin{cases} R - e_{\text{filt}} & i = u \\ 0 - e_{\text{filt}} & i \neq u \end{cases}$$

2.5 Analysis. Maximizing reward stands as the overall goal to gain. Given the structure of the rewards, it is best to maximize the positive reward, R , to optimize action choice. A finite horizon problem can be used to approximate the overall reward such that $\max_{\text{policy}} \mathbb{E}[\sum_{k=0}^{N-1} r]$, with k as an index, N as the number of stages, and no terminal reward.

Through a PBVI algorithm, it is possible to derive an optimal action when given a state, observation, and initial belief. To satisfy the PBVI algorithm, it requires a set of hyperplanes that have been updated to maximize the value function of a subset of belief points. Those belief points must be a representation of the entire belief space. Thus, to solve the problem, a belief update function and an algorithm to maximize the value function must be described. Then a one-step look-ahead problem can be created to execute the algorithms and return an optimal action. The following derivations and equations are in reference to Pineau et al. [13].

2.5.1 Belief Update Function. A belief at some time k can be defined as $\beta_k(i) = \mathbb{P}(S_k = i | H_k)$, where H is the history of observations and actions. The next belief can then be described as $\beta_{k+1}(j) = \mathbb{P}(S_{k+1} = j | H_{k+1})$ and can be written as a function of the

current belief, β_k , action, and observation, $\beta_{k+1}(j) = B(\beta_k, U_k, Y_k)$. This can be expressed in the form

$$\beta_{k+1}(j) = \frac{\sum_i \beta_k(i) P_o P_{tr}}{\sum_i \beta_k(i) P_o} \quad (4)$$

where P_o represents the observation probability and P_{tr} represents the transition probability as defined earlier. Here, β represents a vector of size 1×4 as j can take any value $S_{k+1} \in \{1, 2, 3, 4\}$. So for all j and i combinations, the new β_{k+1} can be created given the old β_k , the action, u , and the observation, y .

For system A, the beta update does not require an action, and will always use the form shown in Eq. (4). System B only chooses Eq. (4) if action 0 is selected as it now has access to the observation signal. For actions $u \in \{1, 2, 3\}$, Eq. (4) can be simplified as $\sum_i \beta(i) = 1$ and the P_o for those actions are 1 when $y = 0$. This simplifies to

$$\beta_{k+1}(j) = \sum_i \beta_k(i) P_{tr} \quad (5)$$

where the next belief represents a 1×4 vector of all j, i pairs—using both Eqs. (4) and (5), the belief update function, $\beta_{k+1} = B(\beta_k, U_k, Y_k)$, can be created for both system A and B.

2.5.2 Approximate Point-Based Value Iteration. To initialize, a Q -dimensional subset of belief points can be determined where $\mathbb{B} = \{\beta^1, \beta^2, \dots, \beta^Q\}$, indexed with β^ℓ . Each β must sum to 1, and can be created either randomly, or by discretization. Random belief points require a larger Q value to maintain the same accuracy, so it is advisable to use a discretized set. The discretized set ensures the space covers all values between 0 and 1 and can be created with $\mathbb{B} = [\frac{a}{Z}; \frac{b}{Z}; \frac{c}{Z}; 1 - \frac{a}{Z} - \frac{b}{Z} - \frac{c}{Z}]$, where $a, b, c \in \{0 : Z\}$ and $(a + b + c) \leq Z$. This creates a subset of belief points of size 1 to Z , where it can be seen later that $Z = 5$ yields a $Q = 56$. For each of the belief points created, an α vector can be associated with it with the subset of relevant hyperplanes as $\mathcal{A}_N = \{\alpha_N^\ell : \ell = 1, \dots, Q\}$ where N is the number of stages. This set can be initialized with $\alpha^\ell = [0; 0; 0; 0]$.

Using a PBVI algorithm gets a sufficiently accurate set of hyperplanes denoted as $\tilde{\mathcal{A}}_0$ by iteratively updating $\tilde{\mathcal{A}}_k$ from $k = N - 1 : 0$. The following equations utilize a PBVI algorithm formulated as a finite horizon problem allowing it to reach 0. The equations can be looped for $k = N - 1 : 0$ to create $\tilde{\mathcal{A}}_0$. To save memory, all past $\tilde{\mathcal{A}}_k$ values can be forgotten and only the final $\tilde{\mathcal{A}}_0$ set needs to be kept.

The max between the inner product of the next belief and α gets the previous value function as seen in Eq. (6). This can then be used to get the current value function in Eq. (7)

$$\tilde{V}_{N-k-1}(B(\beta^\ell, u, y)) = \max_{\alpha \in \tilde{\mathcal{A}}} \langle B(\beta^\ell, u, y), \alpha \rangle \quad (6)$$

$$\tilde{V}_{N-k}(B(\beta^\ell)) = \max_{u \in \mathcal{U}} \sum_i \beta^\ell(i) \left[r_k(i, u) + \sum_y \mathbb{P}(Y_k = y | S_k = i, U_k = u) \tilde{V}_{N-k-1}(B(\beta^\ell, u, y)) \right] \quad (7)$$

The index of the maximum when calculating $\tilde{V}_{N-k}$ yields an optimal action, called u^* . Using the optimal action, the vectors associated with the future value function can be calculated as

$$\alpha_{k+1}^y = \arg \max_{\alpha \in \tilde{\mathcal{A}}} \langle B(\beta^\ell, u^*, y), \alpha \rangle, \forall y \in Y \quad (8)$$

Now the new vectors can be computed as

$$\alpha_k^\ell = \left[r_k(i, u^*) + \sum_{j \in \mathcal{Y}} \mathbb{P}(Y_k = y, S_{k+1} = j | S_k = i, U_k = u^*) \alpha_{k+1}^y(j) \right] i \in \mathcal{S} \quad (9)$$

Equation (9) can be simplified to $\alpha_k^\ell = \left[r_k(i, u^*) + \sum_{j \in \mathcal{Y}} P_o P_{tr} \alpha_{k+1}^y(j) \right] i \in \mathcal{S}$. These updated α vectors can be stored in $\tilde{\mathcal{A}}_k = \{\alpha_k^\ell : \ell = 1, \dots, Q\}$. With each new k , it returns a new $\tilde{\mathcal{A}}_k$ until the final $\tilde{\mathcal{A}}_0$ is reached.

A one step look ahead problem can be used to solve the execution of the algorithm and get the overall optimal action. With $\tilde{\mathcal{A}}_0$ calculated, the following execution can be performed

$$U_k = \arg \max_{u \in \mathcal{U}} \sum_i \beta_k(i) [r_k(i, u) + \sum_y \mathbb{P}(Y_k = y | S_k = i, U_k = u) \max_{\alpha \in \tilde{\mathcal{A}}} (B(\beta^\ell, u, y), \alpha)] \quad (10)$$

Now with a given state, observation, initial β value, and $\tilde{\mathcal{A}}_0$, Eq. (10) can be used to pick the optimal action.

2.6 Simulation. A simulated environment can be created to test the algorithm to simulate a sequence of states and observations. A Markov chain generator can be created to simulate states using the transition probability and initial distribution of $P_0(i) = 0.25$. To provide a uniform distribution, a Gumbel max trick can be used, where $G(i) = -\log(-\log(\text{Uniform}(0,1)))$. Then the index of the max can be chosen as the next state as follows:

$$s_k = \arg \max_{i \in \mathcal{S}} [G(i) + \log(P_{tr})] \quad (11)$$

Given the observation probabilities, an observation can be generated with a generated state. Thus, from the HAR system, 96% of the time, the observation y_k matches the state s_k . Otherwise, it has a uniform chance to become any of the other states $\in \mathcal{S}$. This can be done using real observations from the HAR system and the results from the confusion matrices; however, to simulate thousands of states and observations, the set of observations was simulated using just validation accuracy. Future work can focus on generating observations with the HAR system. This would require a generated sequence of states and a subject performing the state in time. The observed state from the HAR system can be recorded and compared.

Multiple simulations can be run with a simulated environment defined to compare systems A and B. For each test, a thousand states and observations can be simulated. Then, 100 realizations can be done, giving 100 different simulated tests of length 1000. Each realization uses the same simulated sequence of states and observations for every test to compare results fairly. A reward can be calculated based on previously defined reward functions for each state and observation. Then for each realization, a reward value can be calculated to get the average reward per stage, such that $\frac{1}{1000} \sum_{k=0}^{1000} r(s_k, u_k)$. The mean across all realizations can also be calculated to see the average of each.

Systems A and B both use the same belief space, $Q = 56$ and $N = 100$, when calculating the $\tilde{\mathcal{A}}_0$ values before the simulations. Both use a reward, R , of 1, an e_{HAR} cost of 0.5, and an e_{filt} cost of 0.2. An e_{HAR} cost of 0.5 was chosen as there are two main sensor systems, one to create observations of states and one to filter and process PPG signals. Both systems would share the same power source, and since both are Arduino based, they are assumed to pull the same power. With this, a cost of .5 or 50% was chosen. The e_{filt} cost of 0.2 was chosen to account for instances of poorly read data and rare instances that the filter fails.

For every 1000 iterations, an action can be determined using Eq. (10). System A always has access to the observation, y_k ;

however, system B only has access to y_k when an action of 0 is chosen. For all other actions, $u \in \{1, 2, 3\}$, in system B, a $y_k = 0$ must be given to the belief update function in Eq. (10).

Four other policies were created to compare the results from systems A and B. Three of them use the reward setup from system A, and one uses the rewards from system B. The first three are guess policies that choose an action randomly. The first, called off, chooses action 1, no filter, at all times. The second, called even, chooses each action, $u \in \{1, 2, 3\}$, uniformly. The Third, called uneven, chooses action 1 50% of the time, and 2 and 3 25%.

3 Results and Discussion

Figure 5 displays the results comparing the guess policies with system A. The policy that uses system B's rewards is called periodic. At every even-time-step, this system chooses action 0 and gains access to the observations. It then chooses the following action based on the observation. Figure 5 also compares the rewards of the periodic policy and system B.

From Fig. 5, it can be seen that overall, the average cost from system A exceeds just randomly guessing an action, which would make sense as it would accumulate more rewards, R , for its correct guesses. Due to the accelerometer always being on, it finds a negative reward overall as it constantly refers to the HAR system, which begins to outweigh the correct guess. Figure 5 also includes system A; however, it has a much lower reward than the periodic and system B. As the HAR system is selectively referenced, system B can gain a better balance between reward and cost. The periodic system uses the accelerometer every step, lowering its overall reward. It can be seen from Fig. 5 that a higher reward can be achieved when selecting the accelerometer, specifically as in system B. However, the e_{HAR} value was fixed at 0.5, which is about half of the positive reward for a correct guess. To see the effect the e_{HAR} value has on each system, Fig. 5 compares the mean over all 100 realizations for e_{HAR} values from 0 to 1 by 0.2.

By changing the e_{HAR} cost value, it is possible further to visualize the performance of systems A and B. For low accelerometer costs, system A receives higher rewards. This would be due to having full access to the accelerometer at all times and having little to no negative cost due to accessing the HAR system. System A quickly decreases with a linear trend as the cost of e_{HAR} is increased to 1. System B, on the other hand, remains constant regardless of the e_{HAR} value. It can be seen that system B would outperform system A in most cases as the cost of the e_{HAR} has little to no effect on the final outcome of the reward.

From the simulated experiments, it can be seen that system B maintains a higher overall reward when compared to system A. This is due to System B not using the HAR system and avoiding the cost from e_{HAR} for most time steps. The periodic system also outperforms A as it avoids the HAR system's cost since it only half the time refers to the observation. It is also worth noting that the random action choice methods always performed worse for all time points. System B balances correct action choices and references to the HAR system from this.

On the other hand, System A greatly outperforms System B when the cost from e_{HAR} is negligible or zero. This comparison provided insight into the two systems' ability to respond to energy costs in sensor systems. Multisensor systems usually rely on just one power source, which could be costly to run long term. If the cost of using the HAR system were negligible, then System A would be the best option, whereas if the device is energy-constrained, System B will outperform System A.

The POMDP and PBVI framework described shows a system capable of maximizing reward through correct action choice while also providing two distinct Systems, A and B, for use in either energy-constrained or energy-unconstrained devices. For this work, a specific example highlighted the algorithm's ability to work on wearable multisensor systems. The HAR end-to-end example can model the observations, and the action of filter choice was motivated through MA removal for accurate PPG-based HR estimation. What

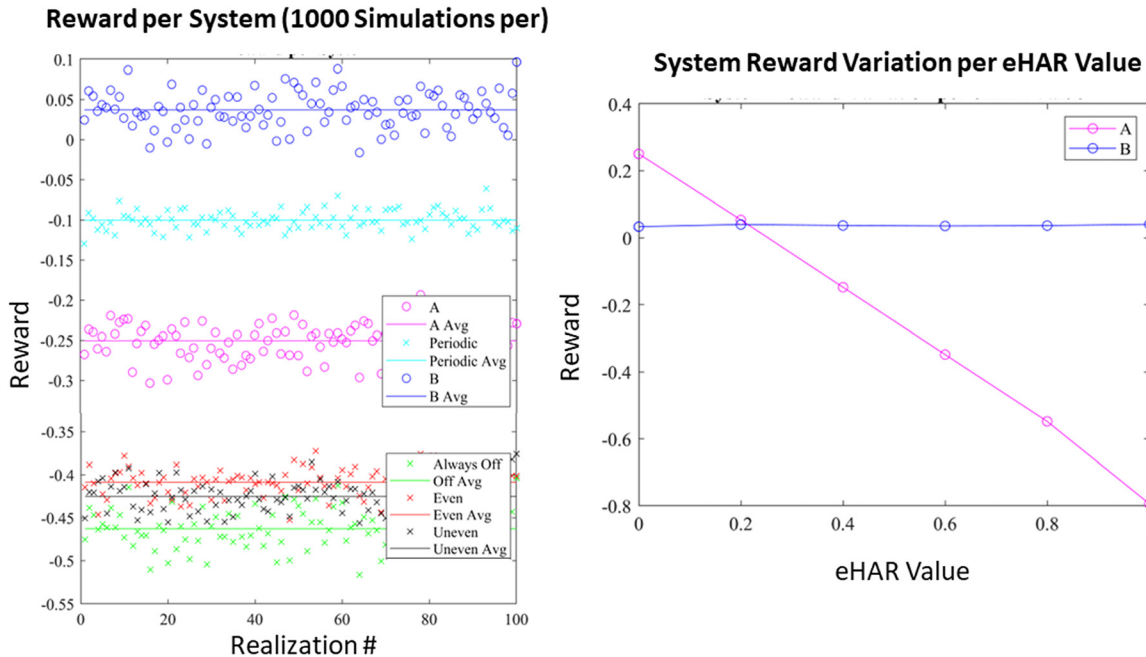


Fig. 5 Reward versus realization (left) and reward by eHAR variation (right) displays the reward per stage. Displays the result of each realization and the average to compare system A, system B, the periodic policy, and the three guess policies. The variation of eHAR is done for systems A and B only.

makes this framework useful is that it can be transferred to almost any similar sensor system problem. To model the observations, a choice can be made on the observation accuracy and can be represented through ϵ . The actions can be altered to represent any desired filter or action a sensor system needs. For example, a new filter can be added to filter sitting and standing. Applying this framework to a wearable assistive device would also be possible where the user requires assistance or device output based on their current activity.

4 Conclusions

This work provides a general framework for optimal action selection on a multisensor or multidevice-based system. A specific example is provided to emphasize the algorithm through transition uncertainty, and action specific filters. The end-to-end embedded HAR system's final accuracy determines a system's action determination. Specific PPG MA filters represent action specific filters which may also have inherit inaccuracies that can be represented as costs to the final system. Using the specific examples, the final system framework can be simulated to find the best action selection method when considering energy constraints. Insights from the application of the POMDP and PBVI framework to a specific multisensor system example can be expanded upon and applied to many novel multisensor systems.

Acknowledgment

This work does not involve human subjects. This work was partially supported by funding from DTRA, CB10787, and NSWC-IDD-NEEC project titled "Development of EWD," Contract No. N00174-20-1-0002, POP 6/1/20-12/31/23, TPOC: Dr. Prabha Dwivedi (e-mail: prabha.dwivedi.civ@us.navy.mil Ph: 404-804-2228), PI: Sangram Redkar (e-mail: sredkar@asu.edu, Ph: 480-727-1129).

References

- [1] Kyriacou, P. A., and Chatterjee, S., 2022, "The Origin of Photoplethysmography," *Photoplethysmography*, A. John and K. Panicos, eds., Academic Press, Cambridge, MA, pp. 17–43.

- [2] Mejia-Mejia, E., Budidha, K., and Allen, J., 2021, "Photoplethysmography Signal Processing and Synthesis," *Photoplethysmography*, Academic Press, Cambridge, MA.
- [3] Wang, M., Li, Z., Zhang, Q., and Wang, G., 2019, "Removal of Motion Artifacts in Photoplethysmograph Sensors During Intensive Exercise for Accurate Heart Rate Calculation Based on Frequency Estimation and Notch Filtering," *Sensors*, **19**(15), p. 3312.
- [4] Ram, M. R., Madhav, K. V., Krishna, E. H., Komalla, N. R., and Reddy, K. A., 2012, "A Novel Approach for Motion Artifact Reduction in PPG Signals Based on as-LMS Adaptive Filter," *IEEE Trans. Instrum. Meas.*, **61**(5), pp. 1445–1457.
- [5] Peng, F., Zhang, Z., Gou, X., Liu, H., and Wang, W., 2014, "Motion Artifact Removal From Photoplethysmographic Signals by Combining Temporally Constrained Independent Component Analysis and Adaptive Filter," *BioMed. Eng. OnLine*, **13**.
- [6] Tanweer, K. T., Hasan, S. R., and Kamboh, A., 2017, "Motion Artifact Reduction From PPG Signals During Intense Exercise Using Filtered X-LMS," 2017 IEEE International Symposium on Circuits and Systems (ISCAS), Baltimore, MD, May 28–31, pp. 1–4.
- [7] Bacà, A., Biagetti, G., Camilletti, M., Crippa, P., Falaschetti, L., Orcioni, S., Rossini, L., Tonelli, D., and Turchetti, C., 2015, "CARMA: A Robust Motion Artifact Reduction Algorithm for Heart Rate Monitoring From PPG Signals," 2015 23rd European Signal Processing Conference (EUSIPCO), Nice, France, Aug. 31–Sept. 4, pp. 2646–2650.
- [8] Kumar, A., Komaragiri, R., Kumar, M., and Pankaj, 2022, "Reference Signal Less Fourier Analysis Based Motion Artifact Removal Algorithm for Wearable Photoplethysmography Devices to Estimate Heart Rate During Physical Exercises," *Comput. Biol. Med.*, **141**, p. 105081.
- [9] Pollreisz, D., and TaheriNejad, N., 2022, "Detection and Removal of Motion Artifacts in PPG Signals," *Mobile Networks Appl.*, **27**(2), pp. 728–738.
- [10] Thatte, G., Li, M., Lee, S., Emken, B. A., Annavaram, M., Narayanan, S., Spruijt-Metz, D., and Mitra, U., 2011, "Optimal Time-Resource Allocation for Energy-Efficient Physical Activity Detection," *IEEE Trans. Signal Process.*, **59**(4), pp. 1843–1857.
- [11] Zois, D.-S., Levorato, M., and Mitra, U., 2013, "Energy-Efficient, Heterogeneous Sensor Selection for Physical Activity Detection in Wireless Body," *IEEE Trans. Signal Process.*, **61**(7), pp. 1581–1594.
- [12] Amiri, D., Anzanpour, A., Azimi, I., Levorato, M., Liljeberg, P., Dutt, N., and Rahmani, A. M., 2020, "Context-Aware Sensing Via Dynamic Programming for Edge-Assisted Wearable Systems," *ACM Trans. Comput. Healthcare*, **1**(2), pp. 1–25.
- [13] Pineau, J., Gordon, G. J., and Thrun, S., 2011, "Anytime Point-Based Approximations for Large POMDPs," *J. Dyn. Sys. Meas. Control.*, **133**(3), pp. 031011-1–031011-2.
- [14] Edge Impulse Inc., 2023, "Edge Impulse," Edge Impulse, San Jose, CA, accessed Mar. 2023, <https://www.edgeimpulse.com/>
- [15] NONIN, 2023, "NONIN WristOx2," Nonin Medical Inc, Plymouth, MN, accessed Mar. 2023, <https://www.nonin.com/products/3150-usb/>
- [16] Shimazaki, T., Hara, S., Okuhata, H., Nakamura, H., and Kawabata, T., 2014, "Cancellation of Motion Artifact Induced by Exercise for PPG-Based Heart Rate Sensing," 2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Chicago, IL, Aug. 26–30, pp. 3216–3219.