Cai and Venkatasubramanian *EURASIP Journal on Information Security* (2018) 2018:13
https://doi.org/10.1186/s13635-018-0082-y

EURASIP Journal on Information Security

RESEARCH Open Access

Detecting data manipulation attacks on physiological sensor measurements in wearable medical systems

Hang Cai and Krishna K. Venkatasubramanian*

Abstract

Recent years have seen the emergence of wearable medical systems (WMS) that have demonstrated great promise for improved health monitoring and overall well-being. Ensuring that these WMS accurately monitor a user's current health state is crucial. This is especially true in the presence of adversaries who want to mount *data manipulation attacks* on the WMS. The goal of data manipulation attacks is to alter the measurements made by the sensors in the WMS with fictitious data that is plausible but not accurate. Such attacks force clinicians or any decision support system AI, analyzing the WMS data, to make incorrect diagnosis and treatment decisions about the patient's health.

In this paper, we present an approach to detect data manipulation attacks based on the idea that multiple physiological signals based on the same underlying physiological process (e.g., cardiac process) are inherently related to each other. We capture the commonalities between a "target" sensor measurement and another "reference" sensor measurement (which is trustworthy), by building an *image reconstruction-based classifier* and using this classifier to identify any unilateral changes in the target sensor measurements. This classifier is *user-specific* and needs to be created for every user on whom the WMS is deployed. In order to showcase our idea, we present a case study where we detect data manipulation attacks on electrocardiogram (ECG) sensor measurements in a WMS using blood pressure measurement as reference. We chose ECG and blood pressure—in arterial blood pressure (ABP) form—because both are some of the most commonly measured physiological signals in a WMS environment. Our approach demonstrates promising results with above 98% accuracy in detecting even subtle ECG alterations for both healthy subjects and those with different cardiac ailments. Finally, we show that the approach is general in that it can be used to build a model for detecting data manipulation attacks that alter ABP sensor measurements using the ECG sensor as reference.

Keywords: Wearable medical systems, Data manipulation attacks, Data-centric attack detection

1 Introduction

Emerging wearable medical systems (WMS) are revolutionizing the way of seeking and delivering healthcare. They have already shown great potential for significantly improving the quality of healthcare and well-being. Figure 1 shows the architecture of a typical WMS environment. It consists of a number of wireless sensing devices (henceforth referred to as *sensors*) which form a distributed wireless network [1] around the body of the person who wears them (i.e., the *user*). These sensors continuously monitor various types of health information from the user and wirelessly communicate this to a *base station*. The base station processes the measured health information, displays them to the user, and may relay them to a medical cloud for long-term storage and for access by caregivers and/or any medical decision support AI. The biggest advantage of using WMS for healthcare and well-being is that it removes all spatio-temporal constraints of traditional healthcare. An individual's health can now be monitored at all times thus providing them with improved quality of care.

The fact that WMS collect sensitive medical data about their users makes them attractive targets for potential adversaries. WMS are safety-critical systems because of their ability to influence treatment. Any attack on them has the potential to cause severe harm to user safety. Lack of security in WMS can have two main consequences: (1)

*Correspondence: kven@wpi.edu
Worcester Polytechnic Institute, 100 Institute Road, Worcester 01609, MA, USA

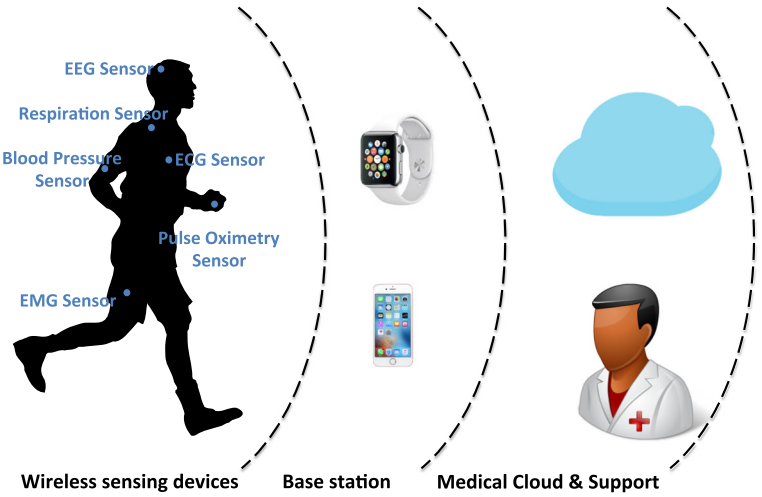

Fig. 1 WMS environment

exfiltration of sensitive health data affecting user privacy and (2) malfunction of the WMS system leading to user harm. One of the ways of causing user harm is through *data manipulation attacks* on sensor measurements. Data manipulation attacks are a type of integrity attack, where an adversary aims to modify the data measured by the physiological sensors in the WMS by targeting and gaining control of the wearable sensors within the WMS system or changing the sensor readings during transit within the WMS by attacking the wireless links.

Recent years have seen a plethora of data manipulation attacks on sensor measurements that go beyond exploiting the open wireless communication channel vulnerability identified for pacemakers [2] and insulin pumps [3]. For instance, sensors are susceptible to a whole class of sensory-channel threats that involve interfering with the transducers of the sensors and introducing arbitrary sensor measurements into the system. This bad-data-injection can be performed using a variety of stimuli including electromagnetic induction [4, 5], light [6–8], and acoustic waves [9, 10]. Such sensory-channel attacks can not only be used to tamper with the sensor measurements [5], but also enable arbitrary code execution under specific conditions, as we ourselves identified [4]. Devices/sensors in WMS have also been compromised by leveraging the fact that they do not typically authenticate the received software and libraries during on-field updates. A compromise of the manufacturer's servers can thus be used to compromise sensors during firmware updates [11]. Finally, adversaries can also physically compromise sensors in the wearable system and subsequently modify their firmware. A variant of this attack is replacing a legitimate device with a malicious one. Attacks on fitness monitors like Fitbit being loaded with malware through open Bluetooth ports is an example of such attacks [12].

In this paper, we present an approach to detect data manipulation attacks on physiological sensor measurements in a WMS environment. The problem of detecting data manipulation attacks on sensor measurements has some similarities with the problem of faulty sensor measurement detection. However, detecting measurements modified by an adversary is a much more challenging task. This is because (1) the adversary may manipulate the measurements in such a way that the modified measurements are erroneous but still clinically plausible. For example, in [5], the authors demonstrated the ability to generate forged electrocardiogram (ECG) measurements by attacking the sensor, allowing the adversary to misrepresent a potentially dangerous arrhythmia in the user's heart and report it as being normal. (2) Traditional fault detection approaches rely on the redundancy of the wireless sensors of the same type, which might not work when we consider physiological signals, as typically there is only one physiological sensor of a particular type in WMS.

Our approach to detect such data manipulation attacks on physiological sensors therefore leverages the fact that different physiological sensor measurements generated by the same underlying physiological process are inherently correlated, i.e., they share similar features among them. Thus, by capturing the commonalities between a user's "target" sensor measurements and other "reference" sensor measurements, we can build a model that can detect when "target" sensor measurements are unilaterally altered. This approach takes advantage of different types of physiological sensors, which already exist in typical WMS, thus avoiding the need of redundant sensors of the

same type. *In the rest of the paper, we use the terms signals, sensor measurements, and sensor outputs interchangeably.*

To illustrate this, we present a case study that particularly focuses on two physiological signals—ECG and blood pressure—in its arterial blood pressure (ABP) form. For our discussion, we assume ECG as the target and ABP as the reference (In Section 8.4, we demonstrate that our approach also works by assuming ABP as the target and ECG as the reference). The reasons we use ECG and ABP signals for our case study are twofold. (1) ECG and blood pressure signals are some of the most commonly measured physiological signals in WMS environments. (2) Both ECG and blood pressure can be measured non-invasively [13] using wearable sensors. Thus, they are ideal to test our basic idea, which is to show that using a related reference signal, we can detect alteration of a "target" signal.

At this point, it is important to note that we are aware that we are making a strong assumption that the adversary cannot compromise the reference signal. One way to think about the problem is that the reference signal provides a form of redundancy for the target signal, without requiring us to have another sensor measuring the target signal. In our specific case study, we assume that the blood pressure reference signal is collected at the trustworthy base station. This can be easily done by implementing the base station in a watch modality like the amulet system [14]. Worn on the wrist, it can measure an untampered blood pressure measurement, which can be used as reference. In our future work, we plan to relax this assumption and eliminate the need for trustworthiness of the reference signal all together.

To detect data manipulation attacks on ECG measurements, our data manipulation detector works by first capturing the tandem variation of the ECG and ABP sensor measurements and using an image reconstruction-based classifier to extract the inter-relationship between the two signals. In this regard, we create two different classes of *images* that capture (1) the inter-relationship between the *unaltered* ECG and ABP sensor measurements and (2) the relationship between *altered* ECG and *unaltered* ABP measurements. For each class of images, we perform principal component analysis (PCA) to compute a set of principal components (PCs), which form the basis of our classifier. Once the two sets of PCs are in place, for any newly received unclassified ECG and ABP signal snippets, a *test image* is first created based on tandem variation of the two signal snippets. Then, the PCs from each class are used to reconstruct the test image. The distances between the test image and two reconstructed test images are then used by a decision function to determine which class the test image belongs to, based on the amount of reconstruction error we observe. An alert is generated if the ECG signal snippet (or more generally the target signal in our model), used to generate the test image, is deemed altered. The use of the PCs allows us to eliminate the need of feature engineering from the images.

Our detector is *user-specific* and needs to be created for every user on whom the WMS is deployed. Our detector has several *advantages*: (1) It does not require redundant ECG sensors to detect data manipulation attacks on ECG sensor measurements. (2) It can detect the alteration of ECG measurement regardless of the type of alterations (in the temporal or morphological sense) to the signal unlike our previous work on this topic [15, 16]. More details on the difference between the work in this paper and our previous work is discussed in Section 2, where we provide more context on the matter. (3) The approach is also agnostic to the attack itself whether it was mounted on the sensors or on the communication links in the WMS.

Analysis of our detector demonstrates promising results with over 98% accuracy in detecting even subtle alterations in ECG signals within 3 s. The *contributions* of this paper are fourfold: (1) the design of an approach for adversarial alteration detection in ECG measurements, (2) demonstration of the efficacy of the detector using real ECG and ABP data from the MIT PhysioBank database [17], (3) demonstration of the robustness of the approach in the presence of a variety of (simulated) data manipulation attacks, and (4) demonstration of the generality of the overall approach by using it to detect data manipulation attacks that alter ABP measurements using ECG measurements as reference.

The rest of the paper is organized as follows. Section 2 presents the related work. Section 3 discusses the system and threat model along with the problem statement. Section 4 presents the background for ECG and ABP. Section 5 presents the main idea of our approach. Section 6 presents the dataset and metrics used in this work. Section 7 presents the parameter selection process for our approach. Section 8 presents the security analysis. Finally, Section 9 presents the conclusions. In the rest of the paper, we use the terms *subject* and *user* interchangeably.

2 Related work

2.1 Fault detection and device instrumentation-based approaches

The consequence of data manipulation attacks is that sensors produce erroneous output. Most of the existing work on detecting erroneous sensor output has focused on the case of faulty sensor detection. Over the years, researchers have developed numerous solutions in this regard, particularly in the domain of wireless sensor networks [18–22]. However, most of the fault detection schemes are based on two main assumptions: (1) the network has a large number of redundant sensors with identical functionality

and (2) for a given stimulus, the sensors in the same neighborhood should have the same or similar sensed values. Given these assumptions, the approaches cluster the nodes into different "subnets" according to their locations and compare the similarity of the device output with others nearby based on a pre-defined threshold. In recent years, researchers have tried to adapt these redundancy-based methods to the domain of WMS [23–27]. As useful as these solutions are for detecting faults with motion sensors, they do not work when we consider physiological sensors, as typically there is only one physiological sensor of a particular type in a WMS. An interesting approach for detecting medical device misbehavior has been developed by Kevin Fu's team at Michigan, called WattsupDoc [28]. The approach uses a supervised machine-learning model to learn a hospital-based medical device's (a drug compounder) behavior with respect to the amount of current the compounder device draws. However, such an approach would require instrumentation of the power adapter of the medical device, which may not be feasible in the WMS context because they are usually battery powered and not connected to the mains. Consequently, in this project, we plan to develop approaches that detect adversarial manipulation of device output without assuming (1) redundancy of devices or (2) device instrumentation.

2.2 Our prior work

Our own previous work has tackled the issue of data manipulation attacks by focusing on detecting ECG sensor output alteration as a result of data manipulation. *However, we have done this in a limited way.* Basically, ECG measurement has two key characteristics—temporal and morphological—both of which can be manipulated by data manipulation attacks. In our previous work, we developed two separate models to detect the temporal and morphological alterations of ECG measurements using reference signals. In [15], we developed *an ECG temporal alteration detection model*, which captures the alterations of the timing properties (RR interval) of the ECG signal by correlating ABP and respiration signals with the ECG signal, while, in [16], we developed *an ECG morphological alteration detection model* that detected a change in the shape of the ECG signal as a result of data manipulation attacks using only the ABP signal as reference. Both approaches relied on the use of identifying hand-crafted characteristics features between the ECG and ABP signals to learn a supervised learning model for learning the normal behavior of the two signals for a user and then use that model to detect unilateral changes to the timing property or the morphology of the ECG signal. Though similar overall to the case study being presented in this paper, there are fundamental differences between our previous work and the current effort.

- In our previous work, we presented two separate models, one for the temporal and one for the morphological case. Both these models were tuned to work individually and not together. Consequently, the temporal alteration detector required 60 min of user data to train and required 5 min of user data to detect alteration, compared to the 20 min to train and 3 s to detect alteration in the morphological case. Combining the two models in a naive manner would have introduced a detection latency bottleneck from the temporal detector. Consequently, in this paper, we take a different approach altogether, which allows to build a unified model that can detect both temporal and morphological alterations in ECG measurements while requiring only 10-min user's data to train the model and 3-s time to generate an alert (as we shall see later).
- Further, our previous work in [15] and [16] required tedious feature engineering to function. In this work, as we shall see, the use of an image-based classifier and principal component analysis allows to us to learn the features automatically, which is easier to design and deploy.

In summary, even though the fundamental idea of the paper of using a related trustworthy signal to detect data manipulation of a sensor output has been presented before, the detector presented in the case study in this paper itself is fundamentally different, much more general, and more thoroughly evaluated.

3 System model, threat model, and problem statement

3.1 System model

We assume the WMS consists of a number of wearable medical devices (i.e., *sensors*). These sensors are low-capability devices that collect physiological data from the user at regular intervals and forward the data to a highly capable sink entity, which we refer to as the *base station*, for processing and storage. The base station provides a root of trust for our system and is assumed to be not susceptible to attacks. Our proposed detection system is deployed at the base station. Consequently, any alteration of the measurements has to occur at the sensor itself or at the wireless link between the sensor and the base station.

3.2 Threat model

We assume the adversaries mounting data manipulation attacks possess several characteristics. (1) Adversaries can cause data manipulation by targeting the wireless link between sensor and the base station or by compromising the sensor's firmware or library update process as in [11]. (2) The data manipulation attacks are assumed to be *not*

advanced persistent in nature and therefore affect a subset of the sensors in the WMS. (3) The non-advanced persistent nature of the attacks also means that the adversary has no prior information on the user including their past medical history or records.

Data manipulation attacks can be used to alter the sensor measurements in a WMS environment in four general ways: (1) by introducing arbitrary noise to the original sensor measurements, (2) by replaying historical sensor measurements stolen from the user in the past as current measurements, (3) by replacing the actual sensor measurements with clinically irrelevant data, and (4) by replacing the real sensor measurements with measurements belonging to another user. In the case of introducing arbitrary noise, the user and their caregivers will immediately be able to see the noise and can therefore ignore the measurements, and inspect the sensor. Replaying historical sensor measurement would require adversaries to access a user's past medical records, which we assume they do not possess. Replacing the actual sensor measurements with clinically irrelevant data can be detected by using conventional faulty sensor data detection techniques. Consequently, in this paper, we focus on the fourth case, where actual sensor measurements are replaced with measurements belonging to other users. This results in clinically plausible sensor measurements for the victim without the measurements being accurate.

3.3 Problem statement

Our goal is to develop an approach for detecting both alteration of sensor measurement in a WMS using a synchronously obtained measurement of an *inherently trustworthy* reference sensor. *In this regard, we primarily focus on detecting malicious temporal and morphological alterations on ECG sensors using ABP sensor measurements as reference. Further, we also aim to show that our larger idea can be used in detecting the converse as well, where ABP is altered and ECG acts as the trustworthy reference.*

4 Background

As we are focused on showing our data alteration detection approach using ECG and ABP signals, in this section, we provide a short summary of the main characteristics and inter-relationship between them. The following paragraph is summarized from our preliminary work on this topic [15, 16]. Both signals are known representations of the cardiac process and reflect the state of a person's cardiovascular system. Figure 2 shows a sample ECG wave. It consists of what is known as an ECG complex. An ECG complex consists of five components, which are usually labeled P, Q, R, S, and T waves. The P-wave signifies atrial depolarization, which causes the blood to be pushed to the ventricles. The QRS complex is observed during the rapid depolarization of the right and left ventricles, which causes the blood to be pushed out of the ventricles into the lungs and the rest of the body. Finally, the T-wave is produced during the depolarization of the ventricles. The time difference between two R-peaks is known as an *RR-interval*, and it refers to the beat-to-beat variations of the heart and is a measure of a person's heart rate. ABP, on the other hand, is the continuous measurement of blood pressure. Figure 3 illustrates a typical ABP signal snippet. The trough of the signal is the diastolic blood pressure and the peak is the systolic blood pressure. Diastolic troughs occur near the beginning of the cardiac cycle and systolic peaks occur when the ventricles contract and push the blood through the entire body. As ECG and ABP signals are both measures of the cardiac process, both of them track each other. For example, an R-peak in the ECG signal will typically be followed by a systolic peak in the ABP signal (see Fig. 4). This is because both represent the compression of the ventricles that results in the blood being circulated through the entire body. Consequently, any pathologies in the cardiac process that results in an abnormal ECG wave form will also be reflected in the ABP signal [29]. This final observation forms the basis of our data manipulation detector.

5 Approach

From here on, we shall focus on the ECG data manipulation attacks. ABP alteration detection using ECG reference will be covered at the very end, in the interest of avoiding repetition. All the details of the ABP detector remain the same as the ECG detector, except the signals switch.

Figure 5 shows the overview of our detector. As we are interested in identifying alterations of the ECG measurements based on the ABP measurements, it is essential to be able to capture the inter-relationship between the ECG and ABP measurements in tandem. Our detector works in two phases: training phase and detection phase. In the training phase, we first create two classes of *images* that capture (a) the inter-relationship between synchronously measured ECG and ABP measurements from a particular user for whom we are in the process of training our detector and (b) the inter-relationship between ABP measured from the user and ECG measured from several other different users (thus modeling the attack where a user's ECG is replaced with someone else's as part of the data manipulation). We then perform PCA on the two classes of images to generate two sets of PCs. Based on the two sets of PCs, we create a user-specific classifier (i.e., a decision function), which can then be used to determine if a newly received snippet of ECG signal has been altered or not. In the detection phase, the classifier simply makes a decision on any newly received ECG measurements from

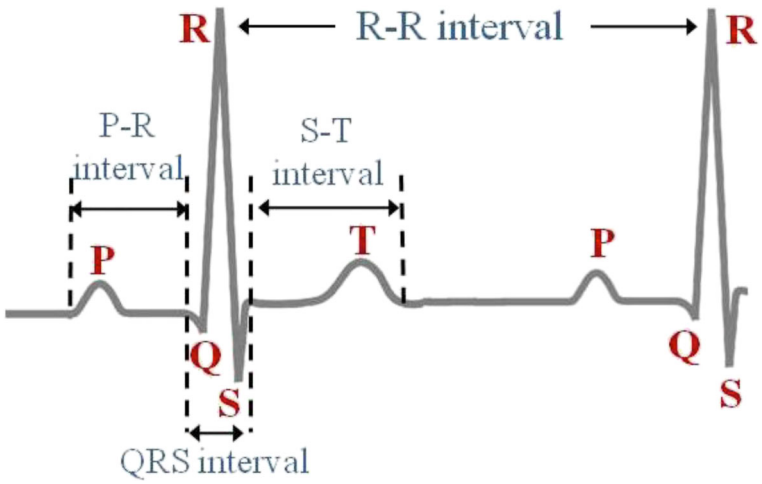

Fig. 2 P, Q, R, S, and T waves in a typical ECG

the user, in conjunction with the synchronously measured, unaltered ABP measurements also obtained from the user.

5.1 Training phase: image creation

In the training phase, we generate *images* of the ECG and ABP signals to capture the inter-relationship between them. These images are generated based on the *portrait* of the two signals. We first describe how to generate a portrait, and we then describe how to generate an image based on the corresponding portrait.

The idea of portrait is inspired by the idea of phase space trajectory, which was originally used to delineate the nonlinear behavior of a dynamic system [30]. A portrait allows us to specify the instantaneous state of several signals over time. We define a *portrait* as an n-dimensional representation of the relationship of several time series in one multi-dimensional space, in this case ECG and ABP signals. To generate a portrait, first, we synchronously measure w time units of ECG and ABP signals. For each w time units of ECG and ABP signals, we then apply unity-based normalization bringing all values to the range [0,1]. Normalization is needed as the magnitude and units of ECG and ABP signals are different. Formally, let $a(t)$ and $e(t)$ be the normalized ABP and ECG signals at time t, where $1 \leq t \leq w$. Then, we create a 2-dimensional portrait, G, using the function $f(t) = (a(t), e(t))$, where again $1 \leq t \leq w$. Figure 6 shows an example of a portrait of ECG and ABP signals.

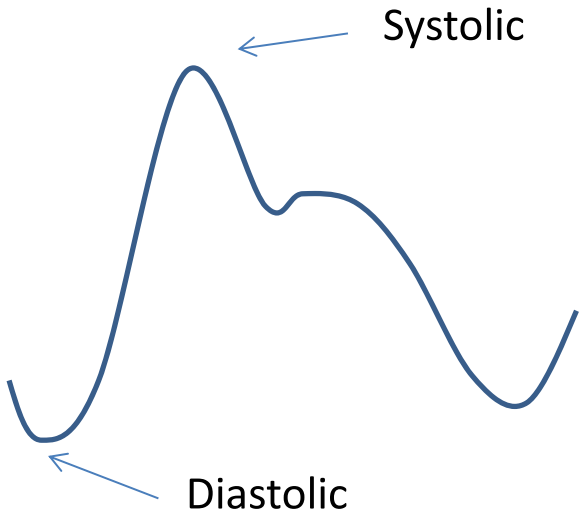

Fig. 3 Systolic peak and diastolic trough in ABP

Once a portrait is created, the next step is to find a way to extract the information from the portrait that captures the inter-relationship between the ECG and ABP signals. To do this, we take a graphical view of portrait at a certain resolution, thus creating an image of the portrait. This image depicts the trajectory formed by ECG and ABP signals in the 2-dimensional space. To create an image I (from portrait G), we first view portrait as an $n \times n$ grid, where each element of the grid records whether there are any points from the portrait that fall into it. We store this information in an $n \times n$ matrix, in which each element $c(i,j)$ is equal to either 0 or 1. The value 1 represents that there is at least one point that falls into the corresponding grid element (i,j). On the other hand, the value 0 represents that there are no points that fall into the corresponding grid element (i,j). In this work, we choose $n = 50$ for generating the matrix, as it allows us to capture the required inter-signal relationship without overly increasing the complexity of resulting image. This binary

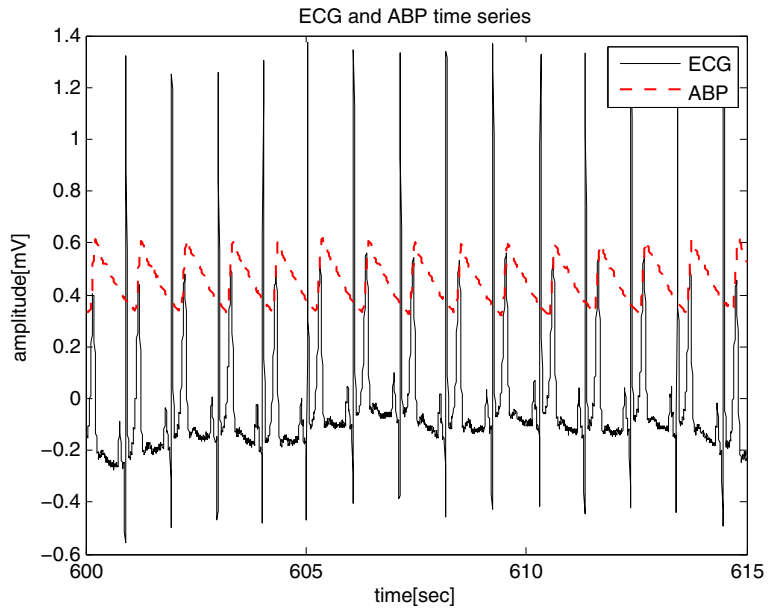

Fig. 4 ECG and ABP superimposed

image matrix with $n \times n$ elements is what we refer to as image I.

Figure 7 shows an example of an image of the portrait of the ECG and ABP signals from Fig. 6. The dark parts are the elements in the image matrix that have a value of 0 and the light parts are those elements that have a value of 1.

During the training phase, we create a user-specific classifier. In this regard, we create two classes of images, one is labeled as negative class and another one is labeled as positive class. The *negative class* images are created by a sliding window of size w, over the synchronously measured ECG and ABP signals from the user whose classifier

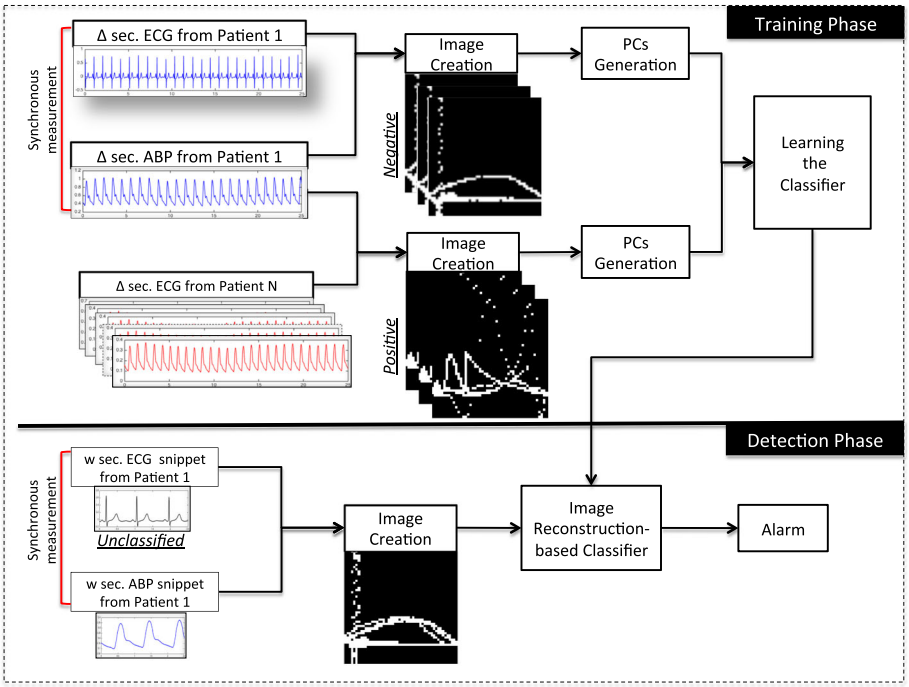

Fig. 5 Detecting malicious alterations of ECG sensor measurements

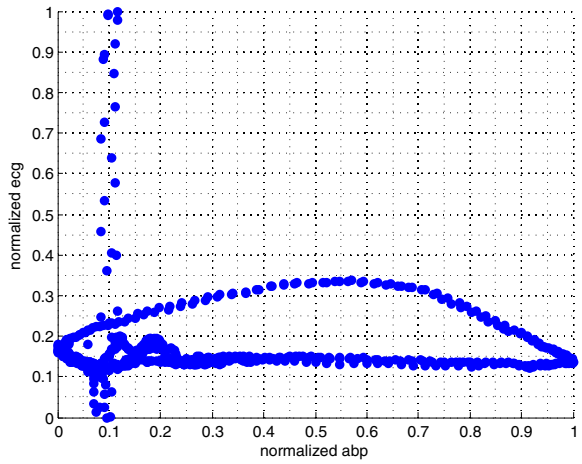

Fig. 6 An example of portrait

we are training. We do this for Δ time units, where $w < \Delta$. The *positive class* images are created by a sliding window of size w, over Δ time units ABP of the user and ECG belonging to several other users. Each w-sized window of data produces one portrait and therefore one image. Consequently, both negative and positive classes have a series of images. Figure 8 shows an example of one negative class image created using synchronously measured ABP $a(t)$ and ECG $e(t)$ from a user, while Fig. 9 shows an example of one positive class image created by using the same ABP $a(t)$ measured from the user and an ECG $e'(t)$ measured from another user, which captures the situations where the ECG of the user is altered with someone

Fig. 7 An example of an image *I* obtained from a portrait *G*

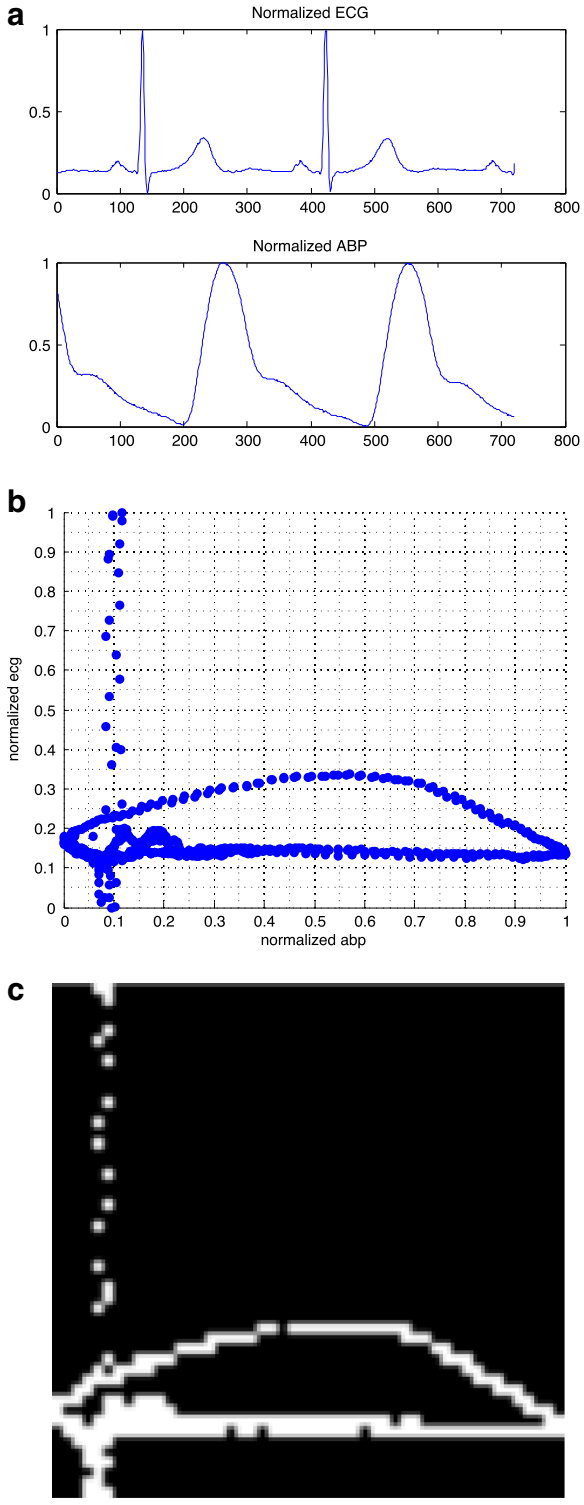

Fig. 8 A negative class image created by ECG and ABP snippets synchronously measured from a user. The top one shows normalized ECG $e(t)$ and normalized ABP $a(t)$. The middle one shows a portrait *G* generated based on $e(t)$ and $a(t)$. The bottom one shows a negative class image *I* generated

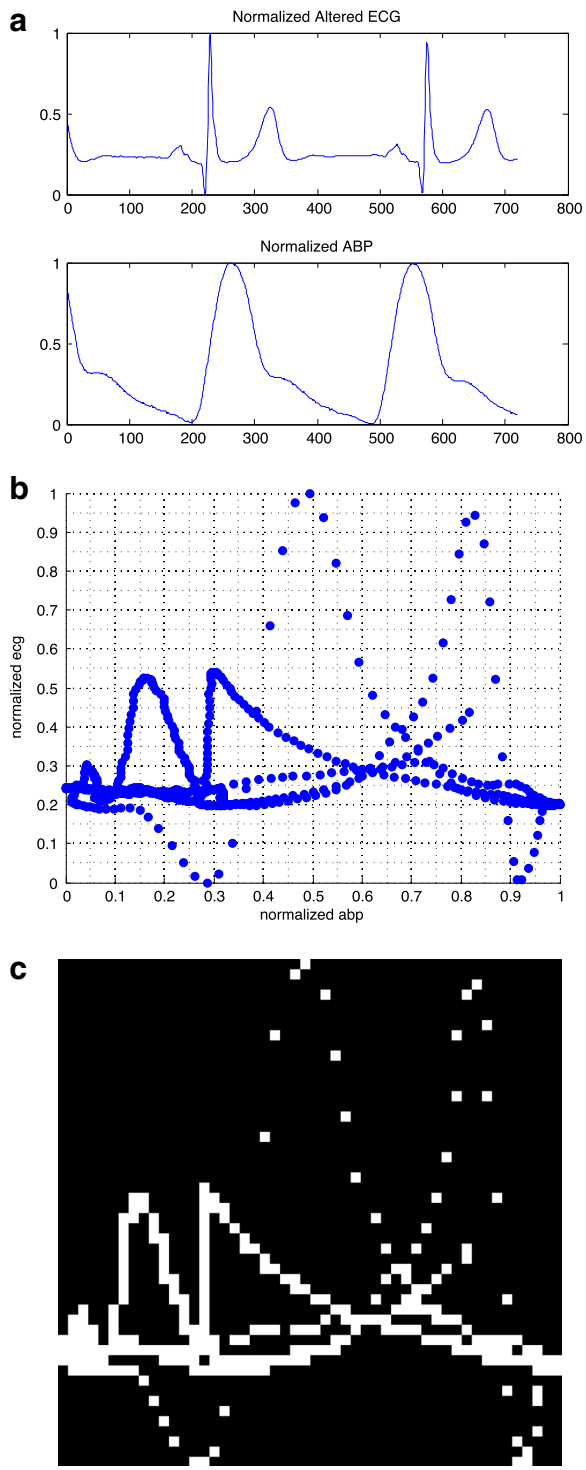

Fig. 9 A positive class image created by the same ABP snippet and an altered ECG measurement. The top one shows normalized altered ECG $e'(t)$ and normalized ABP $a(t)$. The middle one shows a portrait G' generated based on $e'(t)$ and $a(t)$. The bottom one shows a positive class image I' generated

else's through a data manipulation attack. Consequently, the portrait G generated by using the function $f(t) = (a(t), e(t))$ and the portrait G' created by using the function $f(t) = (a(t), e'(t))$ are different, and hence, the image I created based on G and the image I' created based on G' are different.

5.2 Training phase: developing an image reconstruction-based classifier

Once the series of images have been obtained from the training data belonging to both classes, we construct a classifier to identify which class a newly received image belongs to. In this regard, we leverage principal component analysis (PCA), which is a popular technique for compressing the data and has been widely used in many computer vision tasks. From the data analyzed, PCA can derive a set of independent linear combinations of principal components (PCs), which usefully explain variation and bring out strong patterns in the data [31].

Inspired by [32], to build our classifier, we perform PCA on the series of images in both positive and negative classes to obtain two different sets of PCs. The set of PCs generated from the images in a class will preserve important characteristics of that class. Let m be the total number of images in a class, where each image is represented as a column vector v_i with a length of n^2 (as the size of each image is $n \times n$). We then generate a set of *PCs* from these m images in this class, by first creating a covariance matrix C_x such that:

$$C_x = \sum_{i=1}^{m} (v_i - \mu_x)(v_i - \mu_x)^T \tag{1}$$

where x is a label (positive or negative depending upon the class of images for which we are constructing the covariance matrix), and μ_x is the mean of the column vectors of these images in class x which is given by:

$$\mu_x = \frac{1}{m} \sum_{i=1}^{m} v_i \tag{2}$$

The PCs are then generated by finding the eigenvectors of the covariance matrix C_x. Once two sets of PCs have been obtained from the images in each of the class, we essentially have the main element of our classifier. This is because the set of PCs for a given class should capture all the major variations observed in the images of that class. Then, the next step is to classify which class a newly received image u, obtained from yet unseen ECG and ABP measurement snippet, belongs to. We do this by seeing which class PCs can reconstruct the image u the best.

To reconstruct an image using PCs obtained from the images of a given class, we first sort the eigenvectors of the covariance matrix C_x in decreasing order of their corresponding eigenvalues, where x is either the label positive

(pos) or negative (neg). We then select the first k eigenvectors to form a set of PCs and create a matrix P_x. Each row of P_x is an eigenvector obtained from C_x. We can project the image u on this eigenspace as follows:

$$p = P_x(u - \mu_x) \tag{3}$$

We then try to recover the original image from this projection as follows:

$$u'_x = P_x^T p + \mu_x = P_x^T P_x(u - \mu_x) + \mu_x \tag{4}$$

where u'_x is the reconstructed image based on the set of PCs from class x.

Subsequently, we compute the *reconstruction error*, by calculating the Euclidean distance between the reconstructed image (using the set of PCs of a given class) and the original one. We represent the reconstruction error as d_x and calculate it as follows:

$$d_x = \left|u'_x - u\right| \tag{5}$$

Here again, x is the label that can be positive or negative. Finally, based on these two reconstruction errors, we use a *decision function* which outputs the label of this image u as either positive or negative:

$$label(u) = \begin{cases} \text{Positive}, & if \quad d_{pos} - d_{neg} < 0 \\ \text{Negative}, & if \quad d_{pos} - d_{neg} \geq 0 \end{cases} \tag{6}$$

The decision function essentially picks the label of the class whose PCs are better able to recreate the test image and hence have shorter distance, i.e., smaller reconstruction error. The two sets of PCs and the decision function thus form our image reconstruction-based classifier. This classifier is user-specific and needs to be generated for each user individually.

5.3 Detection phase

Once the classifier is in place, we can use it to decide if any newly received snippet of ECG measurements have been maliciously altered or not. In this regard, we collect w time units of newly measured ECG and ABP signals from the user and normalize them to bring all values to the range [0,1]. The normalized w-sized ECG and ABP signals are then used to generate a *test image*. We feed this test image into our user-specific, image construction-based classifier. Our detector first computes two reconstructed images u'_x from this input test image based on Eq. 4 using the two projection matrices P_x and the two means of the column vectors μ_x (obtained during the training phase), where x is either the label positive or negative. Then, by comparing the input test image with its two reconstructed images, respectively, using Eq. 5, two reconstruction errors are generated. Based on these two reconstruction errors, our detector uses its decision function (i.e., Eq. 6) to decide the label of the test image as either positive or negative. If the test image is deemed to be positive, we consider this w second ECG signal snippet to be altered and generate an alert.

6 Dataset and metrics

Before we go into the details of selecting the various tunable parameters of our detectors and measuring its overall performance, we present a short description of the dataset we used for our analysis along with the principal metrics used in our analysis.

6.1 Dataset

In this work, we used data from four databases: MIT PhysioBank Fantasia, MGH/MF, MIT-BIH Normal Sinus Rhythm (NSR), and MIT-BIH Arrhythmia databases [17]. The Fantasia and MIT Normal Sinus Rhythm databases are made up of healthy subjects (i.e., users), while the MGH/MF and MIT-BIH Arrhythmia databases mainly contain data from subjects with specific cardiac abnormalities. As the data source is diverse, the sampling rate of the signals from different databases are not the same. Both Fantasia and MGH/MF databases contain ECG and ABP signals sampled at 250 Hz and 360 Hz, respectively, while MIT-BIH NSR and MIT-BIH Arrhythmia databases only contain ECG signals sampled at 128 Hz and 360 Hz. We upsampled signals from Fantasia and MIT-BIH NSR databases to the same sampling rate, 360 Hz, as our portrait technique requires that both ECG and ABP signals are synchronously measured. We also applied a third-order Butterworth filter with a cutoff frequency at $[1, 50]$ Hz to remove the line-noise and baseline wandering in the ECG data.

We used 33 subjects from Fantasia and MGH/MF databases to form a *training group*, because of the availability of both ECG and ABP signals for them. For each of these 33 subjects, we excluded the first 15 min of their ECG and ABP data as they contain a lot of artifacts and ended up with an average about 41 min of usable ECG and ABP data. These data collected from the subjects of the training group is used to train and validate the classifier. We categorized the subjects in our training group into two types based on their ECG signals: (1) *Normal* subject, which only includes subjects who did not suffer from any ailments and had a normal sinus rhythm ECG, and (2) *Arrhythmia* subject, which only includes subjects who were found to have arrhythmias. In addition, we choose 64 subjects from MIT-BIH NSR and MIT-BIH Arrhythmia databases to form an *external group*. The MIT-BIH NSR and MIT-BIH Arrhythmia databases only provide ECG signals for its subjects; these ECG signals were used to replace the original ECG signals collected from our 33 subjects in the training group, thus simulating a data manipulation attack. The data in the external group also has both normal and arrhythmic ECG data. Table 1 shows

Table 1 Dataset summary

Group	Type	Database	Total . no.	Male	Female	Avg. age (years)	Std. age (years)
Training group	Normal	Fantasia	12	5	7	46.5	25.5
	Arrhythmia	MGH/MF	21	16	5	64	20.3
External group	Normal	MIT-BIH NSR	18	5	13	34.3	8.4
	Arrhythmia	MIT-BIH Arrhythmia	46	26	20	62.6	18.2
All			97	52	45	55.7	21.3

the statistics on the subject population we used to train and test our ECG alteration detector.

6.2 Metrics

We use the following metrics to validate our work: false positive rate, false negative rate, and balanced accuracy rate. We define *false positive (FP) rate* as the fraction of the cases in which an unaltered ECG sensor output is misclassified as altered. Similarly, we define *false negative (FN) rate* as the fraction of the cases where an altered ECG sensor output is misclassified as unaltered. Further, true negative (TN) rate is defined as the fraction of the unaltered ECG sensor output properly classified as unaltered, and true positive (TP) rate is the fraction of the altered ECG sensor output properly classified as altered. We define *balanced accuracy* (BAC) rate as:

$$BAC = 0.5 * TP + 0.5 * TN. \quad (7)$$

The reason we use BAC is that it avoids inflated performance estimates on imbalanced datasets [33]. This is important given that we have an imbalanced sample with many more positive images than negative images during the training phase. Even though we compute these metrics for each subject in our dataset, we validate our approach using summary statistics of these metrics over all subjects.

7 Parameter selection

In this section, we illustrate how we select the three most important parameters of our system: Δ, the amount of (ECG and ABP) data needed to train our classifier (i.e., *training time*); w, the amount of data needed to test for malicious alteration measurements (i.e., *testing time*); and k, the number of principal components we need to build our classifier.

Note that the choice of the three aforementioned parameters is important for the performance of our detector because of the safety-critical nature of WMS. Our system uses w-sized ECG and ABP sensor measurements to decide if the snippet of ECG sensor measurement is altered or not. Therefore, the window size w decides the minimum length of the altered sensor measurement that our system is able to detect. Further, even though the training is done in an offline manner, we need to make sure that it can be done quickly so that the user does not need to endure long interruptions in WMS operation. Consequently, the training time Δ has to be as short as possible as well. Lastly, the number of PCs k determines the number of eigenvectors we choose for our image reconstruction. If k is small, the classifier might not be able to capture enough information of the image during reconstruction. On the contrary, if k is too large, the images reconstructed by two sets of PCs (one for the positive class and one for the negative class) might be too similar to the original image for the decision function to be able to assign them a proper label. Therefore, finding an appropriate value of k is crucial. Figure 10 shows our parameter selection process.

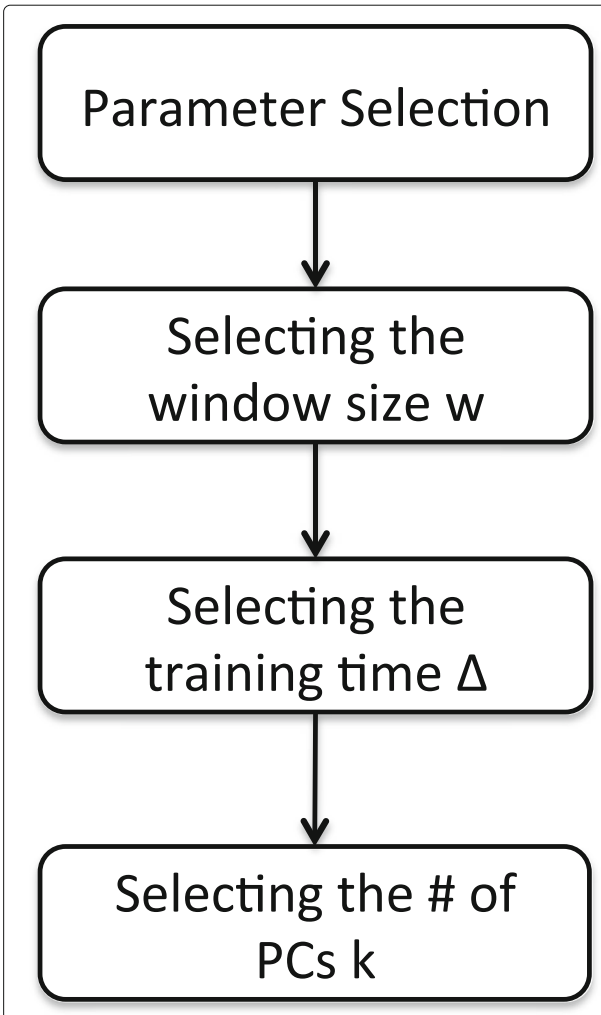

Fig. 10 Process for selecting parameters: window size w, training time Δ, and number of PCs k

To select *window size*, w, we set Δ to a fixed value of 10 min and k to a fixed value of 10 and test our ECG

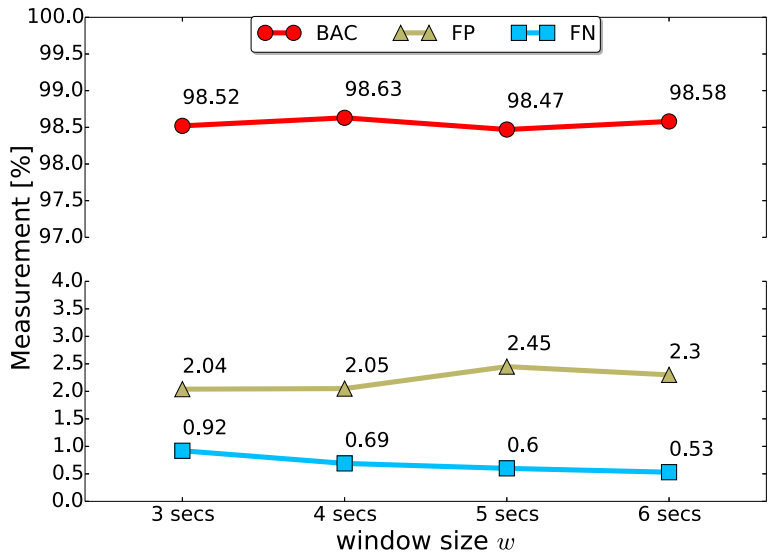

Fig. 11 BAC, FP, and FN rates for various *w*

alteration detector with the data from users in the training group. We tried several values for w and eventually settled on 3, 4, 5, and 6 s. This is because values smaller than 3 s did not capture meaningful information about the inter-relationship between ECG and ABP signals, while values greater than 6 s were assumed too slow for detecting the attacks. For each window size, w, we generated a set of negative class images by using 10 min of synchronously measured ECG and ABP signals from the same subject. We generated the set of positive class images by combining each subject's ABP snippet with a randomly selected ECG snippet from every other subject in the training group. Then, we performed PCA on both positive class images and negative class images to extract two different sets of PCs. The resulting two sets of PCs were then used to train our image reconstruction-based classifier. We used *10-fold cross-validation* to validate each of the classifiers built. Figure 11 shows the average BAC, FP, and FN rates for different window sizes, w. We can see that the average BAC rate of the user-specific classifiers for these four different window sizes w are all considerably high. Overall, we can see that the balanced accuracy rate of classifier when we set w with these four values is largely the same. We therefore choose $w = 3$ s, as it provides us with minimum length of the altered sensor measurement that our system is able to detect. However, if an adversary only altered an ECG snippet less than 3 s, then our detection system may not be able to detect it. This is one of the limitation of our current system, and we plan to reduce this window size w in our future work.

To select the *training time*, Δ, we evaluated the classifier of our ECG alteration detector by training it for four different durations: 5, 10, 15, and 20 min (with a fixed $w = 3$ s and a fixed $k = 10$). Again, for each value of Δ, we generated a set of negative class images and positive class images using data from the training group. Then, we used these two classes of images to train the user-specific, image reconstruction-based classifier for each user in the training group, respectively. We used 10-fold cross-validation to validate each of the classifiers built. Figure 12 shows the average BAC, FP, and FN for different values of Δ. Overall, there is only a slight difference in balanced accuracy rate when we set the training time Δ with four different values. We can see that in terms of the balanced accuracy rate, the training time $\Delta = 10$ and $\Delta = 15$ min are really close (only 0.01% apart) and they outperform the other cases. However, the time that is needed to train the classifier when we set $\Delta = 10$ min is much less compared to the case when we set $\Delta = 15$ min. Consequently, we set $\Delta = 10$ min while training our classifier.

Finally, to select the *number of PCs*, k, we evaluated our ECG alteration detector by training it with four different numbers of PCs k: 5, 10, 15, and 20 (with a fixed window size $w = 3$ s and a fixed training time $\Delta = 10$ min). As before, we first generated a set of negative class images and positive class images using data from the training group. Then, for each value of k, we built our classifier for each user in the training group. We used 10-fold cross-validation to validate each of the classifiers built. Figure 13 shows the average BAC, FP, and FN rates for different values of k. Overall, there is only a slight difference in balanced accuracy rate when we set the number of PCs k from 5 to 20. We can see that the balanced accuracy rates

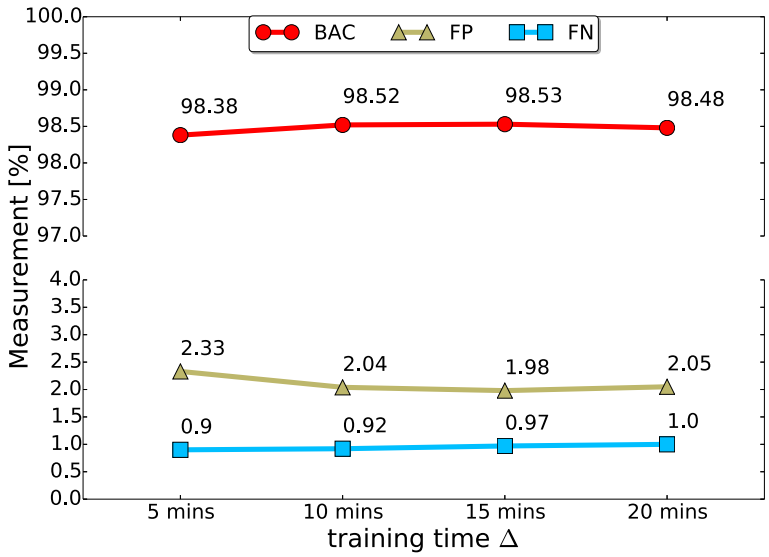

Fig. 12 BAC, FP, and FN rates for various Δ

for $k = 10$ and $k = 15$ are the same and also the highest. We select $k = 10$ for training our classifier as it balances the computational cost of the image reconstruction without reducing the overall accuracy of the classifier. *Therefore, from the above results, we select training time* $\Delta = 10$ *min, window size* $w = 3$ *s, and number of PCs* $k = 10$ *to train the classifier of our ECG alteration detector.*

Figure 14 shows the box plots for balanced accuracy (BAC), false positive (FP), and false negative (FN) rates when we performed 10-fold cross-validation of the classifier of our detector when we set $\Delta = 10$ min, window size $w = 3$ s, and $k = 10$. These results are obtained using each of the 33 user-specific models we have using just the training data (which is determined by the size of Δ), which means using just 10 min of data from our dataset. The purpose of this result is to show how well our model performs when we perform cross-validation on our training data using the 33 users of the training group. As mentioned before, the 33 users in the training group contain users with both normal ECG as well as arrhythmic ECG. We see that the Arrhythmia subject group has a slightly higher spread compared to the Normal subject group with respect to the reported BAC, FP, and FN. This is reasonable as the Normal subject group consists

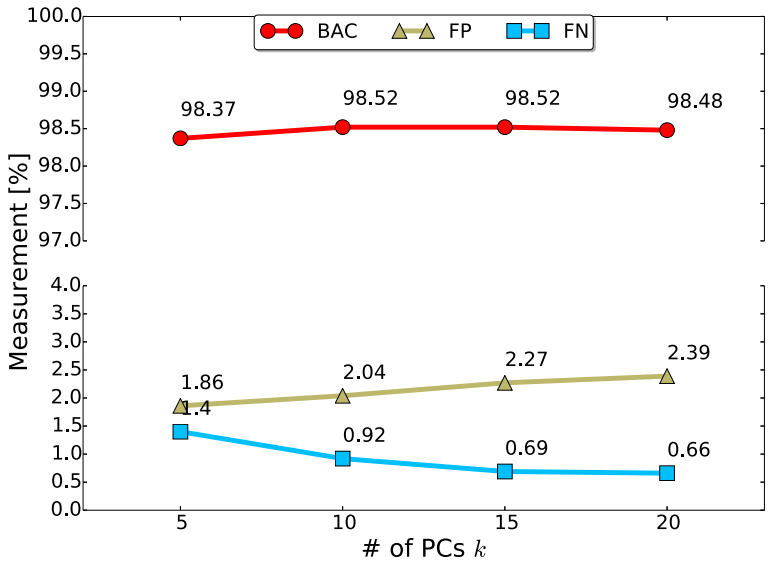

Fig. 13 BAC, FP, and FN rates for various *k*

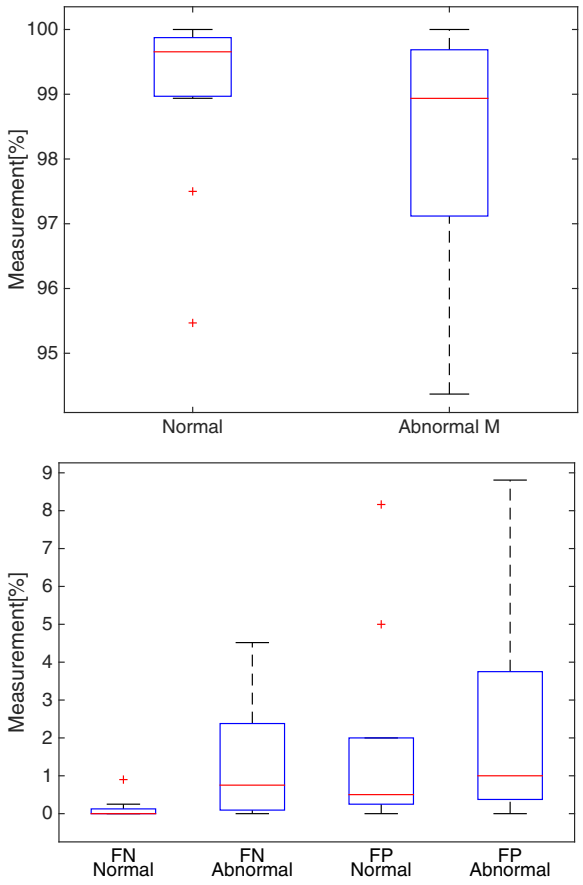

Fig. 14 Validation of ECG alteration detection. The top one shows the validation of ECG alteration detection w.r.t. BAC. The bottom one shows the validation of ECG alteration detection w.r.t. FP and FN rates

only of the subjects with a normal sinus rhythm; on the other hand, each subject in the Arrhythmia subject group has various types of ECG signals. For *Normal* subjects, our detector provides a 99.18% BAC on average with an average false positive rate and an average false negative rate at 1.68% and 0.13%, respectively. Not surprisingly, the performance degrades slightly when we consider subjects with cardiac abnormalities. For *Arrhythmia* subjects, the average BAC rate is 98.19% with an average false positive rate and an average false negative rate at 2.25% and 1.38%, respectively. Overall, the validation results show that the classifier of our ECG alteration detector trained is very accurate with a 98.52% BAC on average with an average false positive rate of 2.04% and an average false negative rate of 0.92%.

In this section, we trained a *user-specific* classifier for each subject in the training group. In the next section, we present the security analysis of our approach by simulating a data manipulation attack that results in alteration of ECG measurements using unseen data from the training group (data beyond the 10-min mark used for training our mode in the dataset) as well as data from the external group.

8 Security analysis

In this section, we address the viability of our detector in detecting ECG alterations from data manipulation attacks. We simulate data manipulation attacks on ECG sensor measurements in three ways: (1) by replacing legitimate ECG measurements of a user (referred to as the *victim* when appropriate) with that of another user in the training group, (2) by replacing legitimate ECG of a user with that of users in the external group, and (3) by replacing legitimate ECG measurements of a user with synthetic ECG obtained from a generative model parameterized based on a victim's own ECG data to impersonate them.

Before we delve into the attack details, we introduce two key notations that we repeatedly use in this section. We define T_{learn} as a time interval for which we collected ECG and ABP data from a user to build our detector. The duration of T_{learn} is the same as the training time Δ, i.e., 10 min. T_{curr} is the time interval where the ECG

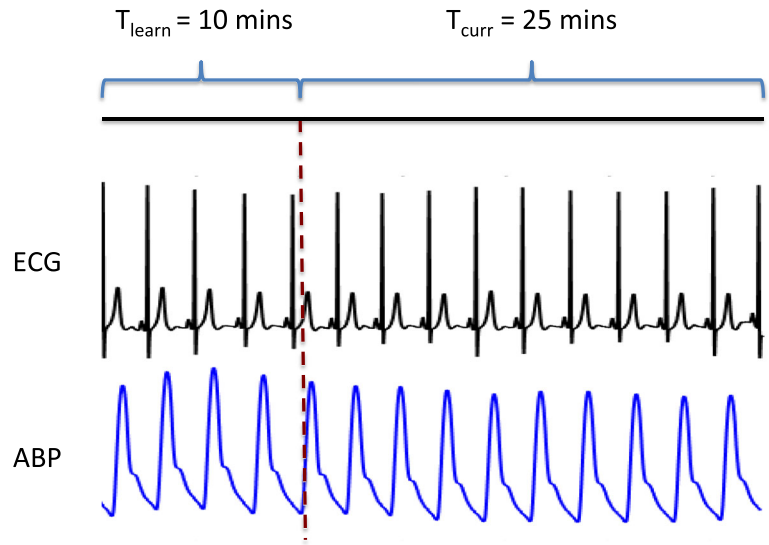

Fig. 15 Timeline for T_{learn} and T_{learn}

measurements of the user are altered. For our analysis, we set T_{curr} to 25 min, and we assume that an adversary can alter any element of the T_{curr} interval of ECG measurements. Note that, since our testing time is only 3 s, as long as the adversary replaces 3 s or more of original ECG measurement, we should be able to detect it. The time durations chosen for T_{curr} is limited by the amount of data we had in our dataset for each user. Figure 15 shows the timeline with T_{learn} and T_{curr}. T_{learn} always precedes T_{curr}.

8.1 Performance in the absence of data manipulation attacks

We first tested our approach to see if it can correctly classify a user's ECG signals (we do not use the term victim in this subsection as there is no attack) once the user-specific detector is trained. This is very important to ensure that our signal alteration detector for a user is able to identify yet unseen data from the same user. Therefore, for each user, we obtained $T_{curr} = 25$ min of synchronous ECG and ABP signals. The two resulting time series are then divided into 500 3-s intervals, each of which produces an image. These 500 images are then fed into the detector, which then labels them as positive or negative. Ideally, we should get all negative labels for the images, as they are from the same user. Overall, when averaged over the 33 user-specific detectors, our approach achieved an average detection accuracy rate of 96.22% (i.e., FP = 3.78%) in detecting unmodified ECG data, which demonstrates that our approach indeed has a low false alarm rate. This result also shows that in most cases, the inter-relationship between ECG and ABP signals did not change over time demonstrating that the detector learned an accurate representation of the user's cardiac process. Further there are 21 male and 12 female users in our training set. The average detection accuracy rate for the detectors built for 21 male users is 95.53% and for 12 female users is 97.42%. Further, the average accuracy rate of the detectors built for 11 adult (18 to 55 years old) users is 97.14% and for 22 senior (56 years old and up) users is 95.76%. These results indicate that our classifiers perform well for a variety of user characteristics in the absence of attacks.

8.2 Performance in the presence of data manipulation attacks

Data manipulation attacks on the ECG sensor measurements require the adversary to replace the victim's ECG measurements with that of another user. The victims here are the 33 users in the training group whose ECG signals are altered. The user-specific models for each of these 33 users (which we also refer to as *victim-specific models*, when justified, for clarity) are the ones that we are evaluating in experiment. All results are consequently reported after being aggregated for all 33 users in the training group. Data manipulation which replaces the victim's ECG with another person's ECG can be mounted in two general ways: (1) by replacing the real ECG measurements from another user in the training group and (2) by replacing the real ECG measurements from another user in the heretofore unseen external group.

First, we consider the case where the adversary modifies T_{curr} duration of a victim's ECG time series with ECG measurements from another user in the training group. This experiment simulates the case where the adversary has access to the ECG measurements of users in the training group (other than the victim whose detector is being evaluated). Thus, for a given user, we replace each of the

consecutive 3-s ECG snippets in his T_{curr} with a randomly selected 3-s ECG measurement snippet obtained from a randomly selected user in the training group. The modified ECG measurement was then fed into the victim-specific detector along with the legitimate (i.e., unmodified) ABP signal measured in T_{curr} from the victim. The detector then produces a label for each 3-s ECG snippet of the modified ECG measurements. In aggregate, our approach achieved an average detection accuracy rate at 99.53% (i.e., FN = 0.47%). Here, we do not have results for false positives because all the 3-s snippets being tested come from someone other than the victim. This demonstrates that even if an adversary has access to the ECG data of users from the training group, our approach can still detect it with considerable accuracy. Further, the average detection accuracy rate for the detectors built for 21 male users is 99.64% and for 12 female users is 99.35%. Lastly, the average accuracy rate of the detectors built for 11 adult (18 to 55 years old) users is 99.89% and for 22 senior (56 years old and up) users is 99.35%. These results indicate that our classifiers perform well for a variety of user characteristics when a user's ECG is replaced with the ECG of another user from the training set.

We now consider the case where the adversary modifies T_{curr} duration of a victim's ECG time series with ECG measurements from another user in the external group. This experiment simulates the case where the adversary replaces legitimate ECG measurements with those of yet unseen users. Thus, for a given victim, we replaced each of the consecutive 3-s ECG snippets in their T_{curr} with a randomly selected 3-s ECG measurement snippet obtained from a randomly selected user in the external group. The modified ECG measurement was then fed into the victim-specific detector along with the legitimate (i.e., unmodified) ABP signal measured in T_{curr} for the victim. The detector then produces a label for each 3-s ECG snippet of the modified ECG measurements. Our approach achieved an average detection accuracy rate at 98.86% (i.e., FN = 1.14%). Again, we do not have results for false positives because all the 3-s snippets being tested come from someone other than the victim. This demonstrates that even if an adversary has access to the ECG data of the users from the external group, our approach can still detect it with considerable accuracy. It is not surprising that the performance of this case is a little worse than the previous case, because the adversary is trying to feed heretofore unseen ECG measurements into the detector. However, the detection accuracy loss is little, demonstrating the robustness of our detector and our approach. Further, the average detection accuracy rate for the detectors built for 21 male users is 98.95% and for 12 female users is 98.70%. Lastly, the average accuracy rate of the detectors built for 11 adult (18 to 55 years old) users is 99.90% and for 22 senior (56 years old and up) users is 98.34%. These results indicate that our classifiers perform well for a variety of user characteristics when a user's ECG is replaced with the ECG of another user from the external set.

8.3 Performance in the presence of data manipulation attacks using synthetic ECG measurements

So far, we have assumed that the adversary does not have access to any information about the user's physiological signal. In this experiment, we loosened this assumption a little. We now consider the case where the adversary has the knowledge of the statistical properties of a victim's ECG signal. This information can be useful for the adversary because they can then use it to generate a synthetic, diagnostically equivalent ECG signal for the victim using generative models for physiological signals. An adversary who has access to a synthetic, diagnostically equivalent ECG signal for a victim can then try to replace their ECG time series with the synthetic ECG obtained from generative detectors parameterized based on that victim's own ECG data.

We use ECGSYN [34], a well-known synthetic ECG generator, which uses a generative model to produce clinically relevant synthetic ECG signals given a set of input parameters. ECGSYN can be parameterized for anyone by collecting their ECG data and extracting certain temporal properties (i.e., represented as average heart rate, standard deviation of the heart rate, and LF/HF ratio. Here, LF stands for low frequency which lies in the range of 0.04 to 0.15 Hz, while HF stands for high frequency which lies in the 0.15 to 0.4 Hz range) and morphological properties (i.e., represented as (a, b, θ), which are the height, width, and distance to R-peak, respectively) from it [34].

To extract both temporal and morphological properties of the victim's ECG, we first collect 5 min of original user's ECG signals measured in T_{learn}. We choose 5 min as that is the minimum amount of ECG signals we need to collect to produce clear waves in both low-frequency and high-frequency bands, which are then used to calculate the one of temporal properties (i.e., LF/HF ratio). We then extract the morphological properties by using curve fitting approach described in [35]. The temporal properties were obtained by detecting the R-peaks, generating RR-interval time series, computing the average and standard deviation of the RR-intervals, and integrating the power-spectral density of RR-intervals over the LF and the HF bands.

Once both temporal and morphological properties are extracted from the original user's ECG, we then use ECGSYN to generate synthetic ECG signals to replace the original user's ECG signals in T_{curr}. Particularly, we consider two different kinds of synthetic ECG signals that the adversary uses to replace the original user's ECG signals: (1) *unaltered synthetic ECG signal*, which has the same temporal and morphological properties with the original

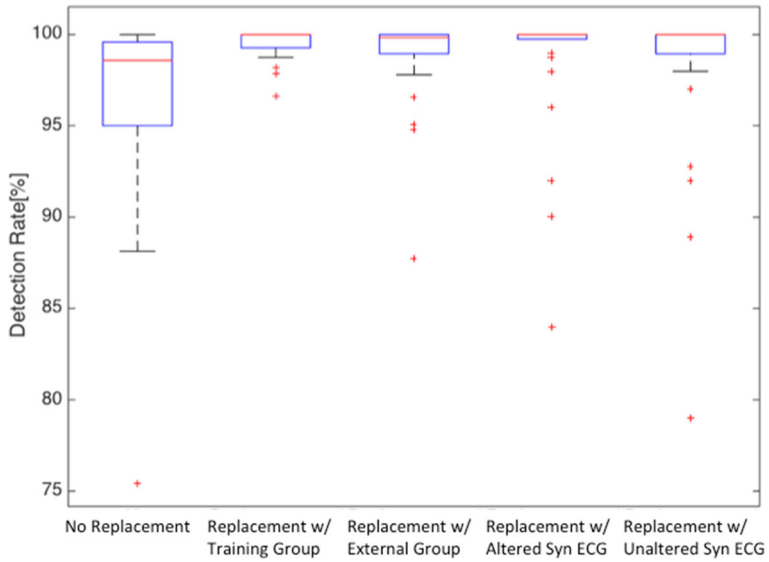

Fig. 16 Summary of detection accuracy for various attack scenarios

user's ECG, and (2) *altered synthetic ECG signal*, which has different temporal properties from the original user's ECG but same morphological properties with the original user's ECG. In the unaltered synthetic ECG signal case, the ECGSYN was parameterized with the temporal and morphological properties of the original user's ECG. In the altered synthetic ECG signal case, the ECGSYN was parameterized with the same temporal and morphological properties of the original user's ECG except with an altered average heart rate parameter to a value outside the usual heart rate range of the user. For instance, if the average heart rate of the victim's ECG is at $[60, 100]$ beats per minute (bpm), we then changed her average heart rate to a random value either above 100 bpm or below 60 bpm. Similarly, if the average heart rate of the original user's ECG is outside of the $[60, 100]$ bpm range, we then changed her average heart rate to a random value between 60 and 100 bpm.

Once the ECGSYN is parameterized, we then generate the two different kinds synthetic ECG signals for each user in the training group. Each synthetic ECG signal time series is 5 min long. We replace each of their randomly selected 3-s ECG snippet in T_{curr} with 3 s of synthetic ECG snippet. The modified ECG measurement was then fed into the victim-specific detector along with the legitimate (i.e., unmodified) ABP signal measured in T_{curr} from the victim. The detector then produced a label for each 3-s ECG snippet of the modified ECG measurements. In aggregate, our approach achieved an average detection accuracy rate at 98.23% (i.e., FN = 1.77%) in detecting unaltered synthetic ECG signal case and an average detection accuracy rate at 98.69% (i.e., FN = 1.31%) in detecting altered synthetic ECG signal case. This demonstrates that even if adversaries have access to the statistics of the original user's ECG signal and induce the synthetic ECG signal based on this information, our approach can still detect it with considerable accuracy.

Further, in the case of detecting unaltered synthetic ECG signal, the average detection accuracy rate for the detectors built for 21 male users is 98.71% and for 12 female users is 97.37%. The average accuracy rate of the detectors built for 11 adult (18 to 55 years old) users is 99.07% and for 22 senior (56 years old and up) users is 97.81%, while in the case of detecting altered synthetic ECG signal, the average detection accuracy rate for the detectors built for 21 male users is 99.05% and for 12 female users is 98.07%. Lastly, the average accuracy rate of the detectors built for 11 adult (18 to 55 years old) users is 100.00% and for 22 senior (56 years old and up) users is 98.04%. These results indicate that our classifiers perform well for a variety of user characteristics when a user's ECG is replaced with the synthetic ECG generated from the parameters of the users themselves.

Figure 16 shows the box plots for detection accuracy rate of our ECG data manipulation detector in detecting different types of alteration of the ECG measurements.

8.4 Performance in the presence of data manipulation attacks on the ABP sensor

Thus far, we built an ECG alteration detector for each user in the training group, based on the assumption that the ABP measurements are trustworthy. Therefore, a natural

question to ask at this point is can the signals be switched. That is, if the ECG measurements are trustworthy, can we build a detector for ABP alteration by leveraging the relationship between ECG and ABP measurements using the exact same approach? To test this hypothesis, once again we set $T_{learn} = 10$ min, $w = 3$ s, and $k = 10$ as the parameters to train a victim-specific ABP alteration detector for each user in the training group. We then used 10-fold cross-validation to evaluate the efficacy of the detector built. We find that the ABP alteration detector had an average BAC rate of 96.63% with a FP rate of 6.04% and FN rate of 0.71%. We also evaluated each victim-specific ABP alteration detector based on its ability to identify (1) yet unseen data from the same user and (2) yet unseen data from another user in the training group. Overall, when averaged over 33 detectors, we achieved an average BAC rate of 93.96% (i.e., FN = 6.04%) in detecting unmodified ABP data and 99.29% (i.e., FN = 0.71%) in detecting alteration of user's ABP with ABP measurements from another user in the training group. We do not provide results for the external group, as in the ECG alteration case, because our dataset does not have ABP data for the subjects in the external group. This demonstrates that, by leveraging the fact that multiple physiological signals based on the same underlying physiological process (e.g., cardiac process) are inherently related to each other, we can build a detector to detect the alteration of a particular physiological signal by using another related physiological signal that is trustworthy. Table 2 summarizes the performance of our ECG data manipulation detector in detecting different alterations of the ECG and ABP measurements.

8.5 Performance comparison with previous work

As mentioned in Section 2, our own previous work has tackled the issue of data manipulation attacks *in a limited way* by focusing on detecting ECG sensor output alteration as a result of data manipulation. We developed two separate models to detect the temporal and morphological alterations of ECG measurements using reference signals. Table 3 shows how our approach is (i.e., image reconstruction-based detector) in comparison to our previous approach (i.e., ECG temporal and morphological alteration detectors). We can see that with respect to the no replacement case, our current approach performs a little bit worse than our previous work. However, with respect to the situation where the ECG sensor measurement is altered with other subjects' ECG, our current approach has a much higher detection rate. Most importantly, this performance is achieved without feature engineering nor characteristic feature (peaks) detection. On the other hand, both our previous works [15, 16] rely on (1) the need of the tedious and sometimes extremely hard feature engineering process and (2) the presence of peak detection algorithms or annotation files for locating the characteristic features. When such peak detection algorithms or annotation files are not available, the performance of our previous approach will definitely be much worse.

Table 2 Summary of security analysis performance

ECG scenarios	Average BAC rate (%)	ABP scenarios	Average BAC rate (%)
No replacement	96.22	No replacement	93.96
Replacement w/ training group measurements	99.53	Replacement w/ training group measurements	99.29
Replacement w/ external group measurements	98.86		
Replacement w/ unaltered synthetic ECG	98.23		
Replacement w/ altered synthetic ECG	98.69		

8.6 Discussion

Note that, in our results, we see that no replacement case (i.e., no attack) fares much worse than the cases where the data manipulation attack actually happens for both ECG and ABP data manipulation detector. This means that our approach is much better at capturing malicious attacks at the expense of creating false alarms. This is not a bad situation to be in for an attack detection system for safety-critical systems. Given the potential extreme consequences of missing a health event due to a data manipulation attack, we believe it is much better to cause a few false alarms which forces the clinicians to look at the patient data when nothing is wrong rather than miss an attack.

In our current design, the classifier is trained in a secure location in an offline fashion with only the alert generation happening online. This requires the user to present in a secure location for training purposes, such as in a hospital in the presence of a medical care provider, before the model being uploaded to the user's base station. Making this training process an online one is one of our future work. Further, our approach relies on the inter-relationship between ECG and ABP signals to operate. If a patient's physiology changes over time, the classifier has to adapt as well. This means that the classifier has to be re-trained every so often in order to capture the current state of the patient's health. One approach is to automate the relearning, based on a schedule. However, choosing the inter-relearning, interval has to be done carefully. Too short an interval would lead to unnecessary relearning and too long an interval would result in increased errors.

Table 3 Comparison with our previous work for detecting data manipulation attacks on ECG sensor measurements

	No replacement (%)	Replacement w/ training group (%)	Replacement w/ external group (%)	Replacement w/ synthetic ECG (%)	Feature engineering	Peak detection algorithm
Temporal detector [15]	97.56	99.35	N/A	91.07	Required	Required
Morphological detector [16]	97.73	90.09	93.79	N/A	Required	Required
Image reconstruction-based detector	96.22	99.53	98.83	98.46	Not required	Not required

Determining the optimal classifier retraining frequency for our work is a user-dependent parameter. For relatively healthy users, the retraining need not happen often, while for individual cardiac conditions, the training has to be done more frequently depending upon the actual condition, how acute it is, and any medications they might be taking. The calculation of optimal classifier retraining is a non-trivial problem in its own right and out of scope for this paper.

Finally, we demonstrated our approach by building (1) *an ECG alteration detection system* using ABP signal as reference and (2) *an ABP alteration detection system* using ECG signal as reference. The success of both two systems in detecting data manipulation attacks indicates that our approach might be applicable on other types of physiological signals. For example, it has been shown that plethysmography and ballistocardiography signals are also highly related to ECG signal [36–39]. In our future work, we plan to build the detection system for other different types of physiological signals to demonstrate our approach is generalizable.

9 Conclusions

In this paper, we presented a novel approach to detect data manipulation-based alteration in WMS environments. Our approach leveraged the idea that if we can capture the inter-relationship between several physiological signals that measures the same underlying physiological process, we can detect a unilateral change in one of them assuming the other signal is not altered and can be used as a reference. In this regard, we focused on the cardiac process and showcased an ECG alteration detector that used ABP measurements. Our detector used an image reconstruction-based classifier to extract the inter-relationship between the ECG and ABP signals and uses it to identify any unilateral changes in the ECG signal. We validated our detector based on the replacement of the actual user's ECG measurements with other subjects' ECG measurements and clinically relevant synthetic ECG signals and demonstrated its (i.e., detector's) efficacy in identifying ECG alteration induced by data manipulation attacks. Analysis of our detector demonstrated promising results with over 98% accuracy in detecting alterations of ECG measurements, within 3 s.

9.1 Future work

In the future, we plan to extend this work in several other directions that are different from what we mentioned before:

- First and foremost, we plan to address the strong assumption in our work of having a trustworthy reference sensor by building a more generalized detector that does not make such assumptions and can detect any signal alteration.
- Further, we plan to work on developing an approach that can help us determine when to retrain our detection models such that we can keep up with the changing physiology of the user, over time.
- We also plan to find optimal ways to alert users into action as a result of detecting data manipulation attacks such that they can take the necessary corrective action to minimize the impact of the attacks. The response time to the alert is also critical in real practice, and it depends on the situation, but the sooner one reacts, the better for the user.
- To train each user-specific detection model, one needs to collect positive class training data from a variety of users. At this stage, it is not clear how many such users are needed to make a generalizable case for our detector. We are obviously limited by the dataset we have access to. In the future, we plan to find ways to determine what is the optimal sample size of our dataset through approaches like statistical power analysis.
- Finally, in this work, we have focused on the developing and analyzing of a classification system that can detect data manipulation attacks. We plan to implement our proposed approach on an actual base station platform (e.g., the amulet system [14]) and evaluate the performance and computational cost.

Abbreviations

ABP: Arterial blood pressure; BAC: Balanced accuracy; FN: False negative; FP: False positive; HF: High frequency; PC: Principal component; PCA: Principal component analysis; TN: True negative; TP: True positive; LF: Low frequency; WMS: Wearable medical systems

Availability of data and materials

All the data used is from four public databases of Physionet: MIT PhysioBank Fantasia, MGH/MF, MIT-BIH Normal Sinus Rhythm (NSR), and MIT-BIH Arrhythmia databases [17].

Authors' contributions
HC made contributions to the design and implementation of the research and wrote part of the paper. KV is HC's advisor and wrote the paper as well. Both authors read and approved the final manuscript.

Authors' information
Hang Cai received a B.S. degree in Automation from Northwestern Polytechnical University, Xi'an, Shaanxi, China, and an M.S. degree in Electrical and Computer Engineering from Worcester Polytechnic Institute, Worcester, Massachusetts, USA. He received his PhD in Computer Science from at Worcester Polytechnic Institute, Worcester, Massachusetts, USA. His research interests include safety and security for wearable medical systems and Internet of Things (IoT), data mining, machine learning, and deep learning.
Krishna K. Venkatasubramanian is an assistant professor at the Computer Science Department of Worcester Polytechnic Institute,Worcester, Massachusetts, USA. He received his Ph.D. in Computer Science from Arizona State University, Tempe, AZ, USA. His research interests include security and fault-tolerance for cyber-physical systems, novel biometrics for wearable/implantable systems, and cyber-security for assistive technologies.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Received: 2 February 2018 Accepted: 21 August 2018
Published online: 29 September 2018

References

1. M. M. Kermani, M. Zhang, A. Raghunathan, N. K. Jha, in *26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems, Pune, India*. Emerging frontiers in embedded security (IEEE, New York, 2013), pp. 203–208
2. D. Halperin, T. Kohno, T. S. Heydt-Benjamin, K. Fu, W. H. Maisel, Security and privacy for implantable medical devices. IEEE Pervasive Comput. **7**(1), 30–39 (2008)
3. C. Li, A. Raghunathan, N. K. Jha, in *2011 IEEE 13th International Conference on e-Health Networking, Applications and Services, Columbia, MO*. Hijacking an insulin pump: security attacks and defenses for a diabetes therapy system (IEEE, New York, 2011), pp. 150–156
4. N. Brown, N. Patel, P. Plenefisch, A. Moghimi, T. Eisenbarth, C. Shue, K. Venkatasubramanian, in *25th Usenix Security Symposium, Austin, TX, 2016*. Poster: SCREAM: sensory channel remote execution attack methods (Usenix Association, Berkeley, 2016)
5. D. F. Kune, J. Backes, S. S. Clark, D. Kramer, M. Reynolds, K. Fu, Y. Kim, W. Xu, in *2013 IEEE Symposium on Security and Privacy, Berkeley, CA*. Ghost talk: mitigating EMI signal injection attacks against analog sensors (IEEE, New York, 2013), pp. 145–159
6. Y. Park, Y. Son, H. Shin, D. Kim, Y. Kim, Korea Advanced Institute of Science and Technology, in *10th Usenix Workshop on Offensive Technologies (WOOT'16), Autin, TX*. This Ain't Your Dose: Sensor Spoofing Attack on Medical Infusion Pump (Usenix Association, Berkeley, 2016)
7. H. Shin, Y. Son, Y. Park, Y. Kwon, Y. Kim, Korea Advanced Institute of Science and Technology, in *10th Usenix Workshop on Offensive Technologies (WOOTŠ16), Autin, TX*. Sampling race: bypassing timing-based analog active sensor spoofing detection on analog-digital systems (Usenix Association, Berkeley, 2016)
8. A. S. Uluagac, V. Subramanian, R. Beyah, in *2014 IEEE Conference on Communications and Network Security, San Francisco, CA*. Sensory channel threats to cyber physical systems: a wake-up call (IEEE, New York, 2014), pp. 301–309
9. R. N. Dean, G. T. Flowers, A. S. Hodel, G. Roth, S. Castro, R. Zhou, A. Moreira, A. Ahmed, R. Rifki, B. E. Grantham, D. Bittle, J. Brunsch, in *2007 IEEE International Symposium on Industrial Electronics, Vigo, Spain*. On the degradation of MEMS gyroscope performance in the presence of high power acoustic noise (IEEE, New York, 2007), pp. 1435–1440
10. R. N. Dean, S. T. Castro, G. T. Flowers, G. Roth, A. Ahmed, A. S. Hodel, B. E. Grantham, D. A. Bittle, J. P. Brunsch, A characterization of the performance of a MEMS gyroscope in acoustically harsh environments. IEEE Trans. Ind. Electron. **58**(7), 2591–2596 (2011)
11. Advisory (ICSA-15-090-03), Hospira MedNet vulnerabilities (2015). https://ics-cert.us-cert.gov/advisories/ICSA-15-090-03/
12. '10-second' theoretical hack could jog Fitbits into malware-spreading mode (2015). http://www.theregister.co.uk/2015/10/21/fitbit_hack/. Accessed 02 Apr 2017
13. The ClearSight system (n.d.) http://www.edwards.com/eu/products/mininvasive/pages/clearsightsystem.aspx. Accessed 02 Apr 2017
14. J. Hester, T. Peters, T. Yun, R. Peterson, J. Skinner, B. Golla, K. Storer, S. Hearndon, K. Freeman, S. Lord, R. Halter, D. Kotz, J. Sorber, in *14th ACM Conference on Embedded Network Sensor Systems (SenSys '16)*. Amulet: an energy-efficient, multi-application wearable platform (ACM, New York, 2014), pp. 216–229
15. H. Cai, K. K. Venkatasubramanian, in *Network and System Security. NSS 2015. Lecture Notes in Computer Science*, ed. by M. Qiu, S. Xu, M. Yung, and H. Zhang. Detecting malicious temporal alterations of ECG signals in body sensor networks, vol. 9408 (Springer, Cham, 2015), pp. 531–539
16. H. Cai, K. K. Venkatasubramanian, in *2016 International Conference on Distributed Computing in Sensor Systems (DCOSS), Washington, DC*. Detecting signal injection attack-based morphological alterations of ECG measurements (IEEE, New York, 2016), pp. 127–135
17. A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, H. E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation. **101**(23), 215–220 (2000)
18. Z. Taghikhaki, M. Sharifi, in *2008 11th International Conference on Computer and Information Technology, Khulna, Bangladesh*. A trust-based distributed data fault detection algorithm for wireless sensor networks (IEEE, New York, 2008), pp. 1–6
19. M. Hajibegloo, A. Javadi, in *2012 Second International Conference on Digital Information and Communication Technology and it's Applications (DICTAP), Bangkok, Thailand*. Fast fault detection in wireless sensor networks (IEEE, New York, 2012), pp. 62–66
20. M.-H. Lee, Y.-H. Choi, Fault detection of wireless sensor networks. Comput. Commun. **31**(14), 3469–3475 (2008)
21. P. Jiang, A new method for node fault detection in wireless sensor networks. Sensors. **9**(2), 1282–1294 (2009)
22. J. Chen, S. Kher, A. Somani, in *2006 workshop on Dependability issues in wireless ad hoc networks and sensor networks (DIWANS '06)*. Distributed fault detection of wireless sensor networks (ACM, New York, 2006), pp. 65–72
23. K. Duk-Jin, B. Prabhakaran, Motion fault detection and isolation in body sensor networks. Pervasive Mob. Comput. **7**(6), 727–745 (2011)
24. A. Mahapatro, P. M. Khilar, Fault diagnosis in body sensor networks. Int . J. Comput. Inf. Syst. Ind. Manag. Appl. (IJCISIM). **5**, 252–259 (2013)
25. D. Kim, M. H. Suk, B. Prabhakaran, in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Diego, CA*. Fault detection and isolation in motion monitoring system (IEEE, New York, 2012), pp. 5234–5237
26. H. Sagha, J. del R Millan, R. Chavarriaga, in *2011 International Conference on Body Sensor Networks*. Detecting and rectifying anomalies in body sensor networks, (2011), pp. 162–167
27. S. Galzarano, G. Fortino, A. Liotta, in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Seoul, South Korea*. Embedded self-healing layer for detecting and recovering sensor faults in body sensor networks (IEEE, New York, 2012), pp. 2377–2382
28. S. S. Clark, B. Ransford, A. Rahmati, S. Guineau, J. Sorber, W. Xu, K. Fu, in *USENIX Workshop on Health Information Technologies*. WattsUpDoc: Power side channels to nonintrusively discover untargeted malware on embedded medical devices, (2013). https://spqr.eecs.umich.edu/papers/clark-healthtech13.pdf
29. Abnormal EKG's and corresponding arterial waveforms (2001). http://www.dynapulse.com/educator/WebCurriculum/Chapter%203/Abnormal%20EKG%20and%20Waveform.htm. Accessed 02 Apr 2017
30. S. M. Krishnan, D. N. Dutt, Y. W. Chan, V. Anantharaman, in *Advances in Cardiac Signal Processing*, ed. by U. R. Acharya, J. S. Suri, J. A. E. Spaan, and S. M. Krishnan. Phase space analysis for cardiovascular signals (Springer, Berlin, 2007), p. 339
31. S. Wold, K. Esbensen, P. Geladi, Principal component analysis. Chemometr. Intell. Lab. Syst. **2**(1-3), 37–52 (1987)
32. L. Malagón-Borja, O. Fuentes, Object detection using image reconstruction with PCA. Image Vis. Comput. **27**(1), 2–9 (2009)

33. D. R. Velez, B. C. White, A. A. Motsinger, W. S. Bush, M. D. Ritchie, S. M. Williams, J. H. Moore, A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction. Genet. Epidemiol. **31**(4), 306–315 (2007)
34. P. E. McSharry, G. D. Clifford, L. Tarassenko, L. A. Smith, A dynamical model for generating synthetic electrocardiogram signals. IEEE Trans. Biomed. Eng. **50**(3), 289–294 (2003)
35. S. Nabar, A. Banerjee, S. K. S. Gupta, R. Poovendran, in *2011 International Conference on Body Sensor Networks, Dallas, TX*. GEM-REM: Generative model-driven resource efficient ECG monitoring in body sensor networks (IEEE, New York, 2011), pp. 1–6
36. G. Lu, F. Yang, J. Taylor, J. Stein, A comparison of photoplethysmography and ECG recording to analyse heart rate variability in healthy subjects. J. Med. Eng. Technol. **33**(8), 634–641 (2009)
37. W. H. Lin, D. Wu, C. Li, H. Zhang, Y. T. Zhang, in *The International Conference on Health Informatics. IFMBE Proceedings*, ed. by Y. T. Zhang. Comparison of heart rate variability from PPG with that from ECG, vol. 42 (Springer, Cham, 2014), pp. 213–215
38. A. Martín-Yebra, et al, in *2015 Computing in Cardiology Conference (CinC), Nice, France*. Studying heart rate variability from ballistocardiography acquired by force platform: comparison with conventional ECG (IEEE, New York, 2015), pp. 929–932
39. M. D. Zink, C. Brüser, P. Winnersbach, A. Napp, S. Leonhardt, N. Marx, P. Schauerte, K. Mischke, Heartbeat cycle length detection by a ballistocardiographic sensor in atrial fibrillation and sinus rhythm. BioMed Res. Int. **2015**, 10 (2015). Article ID 840356