Online Anomaly Detection for Long-Term ECG Monitoring using Wearable Devices

Diego Carrera[a,*], Beatrice Rossi[b], Pasqualina Fragneto[b], Giacomo Boracchi[a]

[a] *Politecnico di Milano, DEIB*
[b] *STMicroelectronics*

Abstract

Many successful algorithms for analyzing ECG signals leverage data-driven models that are learned for each specific user. Unfortunately, a few algorithmic challenges are still to be addressed before employing these models in wearable devices, thus enabling online and long-term monitoring. In particular, since the heartbeats morphology changes with the heart rate, models learned in resting conditions need to be adapted to analyze ECG signals recorded during everyday activities.

We propose an online ECG monitoring solution where normal heartbeats of each specific user are modeled by dictionaries yielding sparse representations, and heartbeats that do not conform to this model are detected as anomalous. We track heart rate variations by adapting the user-specific dictionary with a set of user-independent, linear, transformations. Our experiments demonstrate that these transformations can be successfully learned from a public dataset of ECG signals and that, thanks to an optimized anomaly-detection algorithm, our solution enables online and long-term ECG monitoring.

Keywords: Online and long-term ECG monitoring, anomaly detection, domain adaptation, wearable devices, sparse representations

*Corresponding author
Email address: diego.carrera@polimi.it (Diego Carrera)

1. Introduction

Health and ECG monitoring have attracted attention in the pattern-recognition and machine-learning literature, and algorithms to support the manual annotation of ECG signals have been widely researched to alleviate this expensive and time consuming operation. Many algorithms to automatically detect anomalies and classify heartbeats have been proposed [1, 2, 3, 4, 5, 6, 7, 8], and a few commercial products [9, 10] implementing these features are nowadays available. Often, physicians employ these tools to preliminarily scan long ECG signals and highlight those relevant segments which require careful inspection.

The next frontier is to bring these algorithms directly on sensing devices, easing the transitioning from hospital to home/mobile monitoring. Wearable devices have a huge potential in this scenario [11], since their computational power and sensor suite have steadily improved in the past few years. However, their practical use in real-life applications is far from being straightforward, since a few algorithmic challenges have still to be addressed. In particular, here we consider the following.

Learning user-specific models. Most of successful algorithms adopts data-driven models that are learned from each specific user [12, 7, 13]. In fact, as illustrated in Figure 1, the heartbeats of each user feature a very specific morphology, which also depends on the sensing apparatus and the electrodes position [14]. As such, to successfully classify or detect anomalies in ECG signals, a single data-driven model is not able to accurately describe all the users (even when trained on large datasets), and has to be directly learned / customized from the heartbeats of the specific user [6, 7]. In a wearable-device scenario, where the user places the device by himself, this training / customization of the data-driven model has to be performed every time the device is placed, since heartbeats acquired from the same user and device exhibit a different morphology when the device position changes [15, 14].

Efficient anomaly detection. To raise timely alerts in case of dangerous arrhythmias and avoid massive data-transfer, wearable devices should prelimi-

narily screen the ECG signals while being recorded. In particular, the device has to identify, in an online manner, any anomalous heartbeat which does not conform with the morphology of user's normal heartbeats. Performing heartbeat classification directly on the wearable device is often unfeasible, since in ECG monitoring user-specific and data-driven models are conveniently learned in an unsupervised manner, without requiring a physician to label the ECG signals and examples of anomalous heartbeats. Moreover, many classifiers, such as those based on deep neural networks [7], which could in principle be customized to each user through transfer learning [16], exhibit high computational requirements which are typically not compatible with wearable device resources.

Autonomous adaptation of the learned model. As shown in Figure 1, the heartbeats morphology changes when the heart rate increases. Thus, a model learned from heartbeats acquired in the initial training session (presumably when the user is in resting conditions) might not properly describe heartbeats acquired in long-term recordings during everyday activities. Such a mismatch typically implies that anomaly-detection algorithms raise a large amount of false alarms. Unfortunately, straightforward solutions like learning a model for each heart rate are not viable, since the training procedure has to be kept simple and without risks. Therefore, to avoid performance degradation, user-specific data-driven models have to be adapted online to match heart rate variations.

Here, we propose a solution to perform online ECG monitoring on wearable devices that addresses all these challenges. The data-driven models we adopt are dictionaries yielding sparse representations [18, 19], namely matrices whose columns identify a union of low-dimensional subspaces gathering normal heartbeats of the corresponding user. At each device placement, few minutes of ECG signals are recorded from the user in resting conditions: these are enough to learn a user-specific dictionary. To determine whether an heartbeat is anomalous, we propose a very efficient algorithm to compute the distance between the heartbeat and the union of low dimensional subspaces identified by the learned dictionary. Anomalies are then detected by thresholding this distance, to report any heartbeat that does not conform with user resting conditions.

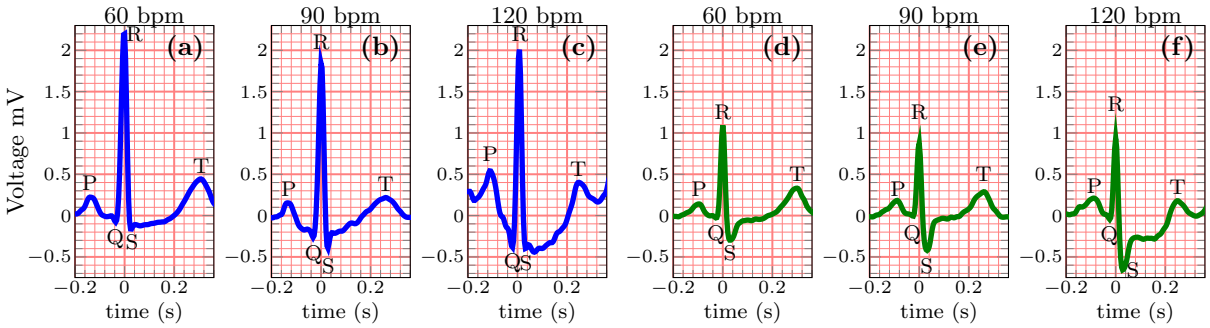


Figure 1: Examples of heartbeats acquired at different heart rates measured in beats per minute (bpm) from two users in the LTSTDB dataset: Figures (**a-c**) refer to user s20071, while Figures (**d-f**) refer to user 20431. Letters indicate the waveforms in an ECG [17], namely the P wave, the QRS complex and the T wave, while the heart rates are reported over each plot. The heartbeats get transformed when the heart rate increases: the T-waves approach the QRS complexes, the QT intervals narrow down and the support of each heartbeat contracts (Figures **b**, **c**, **e**, **f**). These heartbeats from different users undergo a similar transformation, which does not seem to be simple dilation / contraction, since peaks change their intensities and shapes.

We also present a practical solution to adapt both the dictionary and the anomaly-detection threshold when the heart rate changes, thus enabling long-term and online ECG monitoring. Our solution can be seen as a form of *domain adaptation* or *transductive transfer learning* [16], where an algorithm trained on *source domain* data (the ECG signal acquired in resting conditions at low heart rate) is adapted to operate in a different *target domain* (at higher heart rates). We perform adaptation by linear transformations that depend only on the heart rate. Our experiments show that these transformations can be successfully learned from a large, publicly available, dataset containing long ECG signals of several users, like the ones in [20]. This result is perhaps justified by the fact that different human hearts feature the same electrical conduction system, thus we can reasonably assume that, while the morphology of the heartbeats is user specific, the heartbeats get transformed in the same way for all the users.

We have carefully optimized our solution to make it compatible with the limited resources available on the Bio2Bit Dongle [21], a prototype wearable device developed by STMicroelectronics. In particular, we present an efficient variant of the anomaly-detection algorithm that exploits the algebraic proper-

ties of the learned low dimensional subspaces. Our domain-adaptation solution can be seamlessly integrated in the online monitoring without increasing the computational complexity of the anomaly detection algorithm.

This paper extends [15], which presents the algorithm for detecting anomalous heartbeats, [22] which introduces dictionary adaptation and [21] which describes the ECG monitoring device, in the following direction: *i*) the adaptation of the decision rule w.r.t. the heart rate, to make our solution able to perform long-term monitoring, *ii*) an experimental campaign over long-term recordings of more than 200 patients from publicly available datasets and 15 ECG signals acquired using the Bio2Bit Move, and *iii*) a deeper analysis of computational complexity and battery life of a device running the proposed solution. Moreover, we present a unified problem formulation including both the anomaly detection and domain adaptation, and extend the review of the related literature.

The paper is structured as follows. Section 2 overviews related works, while Section 3 provides a formal statement of the anomaly-detection and domain-adaptation problems. The anomaly-detection algorithm is described in Section 4, and our domain-adaptation solution is in Section 5. Section 6 details the optimized anomaly detection algorithm to be run on the Bio2Bit Move device, and describes the user-configuration phase that to be run every time the device is placed. Section 7 presents our experimental campaign, and conclusions are drawn in Section 8.

2. Related Works

Since the introduction of Holter devices in 1940s, cardiac monitoring has helped physicians to determine whether users are experiencing anomalous heartbeats. Over the past few years, ECG monitoring devices have evolved from large, wired systems, as the original Holter, to small, wire-free wearables that allow users to perform everyday activities with minimal disturbance. Most of ECG monitoring devices typically do not implement on board anomaly-detection and heartbeat-classification functionalities, and ECG signal are sensed, processed and transferred to caregivers that remotely monitor the users [11, 23]. The

challenge we address here is to integrate anomaly-detection capabilities directly on a low-power wearable device, as this could raise timely alarms and prevent massive data-transfer that would reduce the device battery lifetime.

Pattern recognition and machine learning techniques have been widely exploited for anomaly-detection and heartbeat-classification purposes. A first class of solutions addressing these tasks are the so called *feature-driven* methods [24, 5], which exploit hand-crafted morphological features that mimic the criteria used by clinicians to analyze ECG signals. Typical examples of features are the RR interval, namely the distance between two consecutive R-peaks, the amplitude and the width of QRS complex, as well as shape descriptors for the local waveforms like P-wave, T-wave, and ST-segment (see Figure 1). Other features can be extracted from the vectorcardiogram [25], computed through a linear transformation of the 12 leads ECG, or in wavelet domain [1] or through Hermite transform [12]. In the last few years, feature-driven approaches are being replaced or combined with *data-driven* methods, which do not reproduce any clinical criterion, but are directly learned from training data. Data-driven methods typical leverage a model yielding meaningful representations of the ECG signals [2, 6, 7, 15], and often refer to the time series literature [3, 8] and detect anomalies by monitoring the prediction error. Others resort to clustering [6, 26] or Gaussian Mixture Models [3, 2] to describe normal heartbeats. The main drawbacks of these methods is that they process the global ECG signal offline, and are not suitable for online monitoring. Recently, deep neural networks have shown very good performance in heartbeats classification [7, 27] and the 1-dimensional Convolutional Neural Network (CNN) in [7] reaches an accuracy of 98.6% on 24 recordings of the MIT-BIH database. The deep Long Short Term Memory (LSTM) network in [28] detects anomalies in time series where the pattern duration is unknown. Unfortunately, the computational requirements of deep neural networks are not compatible with the limited resources available on most wearable device, such as the Bio2Bit Dongle: the CNN in [7] classifies a single heartbeat in few milliseconds on a 2.4 GHz CPU with 16 GB of RAM, while the CPU frequency of most wearable devices is in the order of tens of

MHz.

As discussed in Section 1, to successfully perform long-term ECG monitoring it is necessary to adapt the learned model to track heart-rate variations. This problem has been so far ignored in ECG monitoring, including [29, 30], which are examples of ECG monitoring algorithms meant for wearable devices. Our adaptation problem calls for the *transfer learning* scenario [16], in which a model learned in the source domain is transformed to operate in a target domain where observations come from a different distribution. The most often transferred task is perhaps classification, and the goal is to learn, from labeled data in the source domain, a classifier that can operate also in the a target domain, where a few supervised (or possibly unsupervised) data are provided [31, 32]. Since in ECG monitoring, training data of each user are available only in the source domain, it is more appropriate to specifically refer to *domain-adaptation* or *transductive transfer learning* methods [16].

In this paper we consider dictionaries yielding sparse representations as data-driven models to detect anomalous heartbeats, and describe a practical domain-adaptation solution that allows monitoring at different heart rates. Dictionaries yielding sparse representations are one of the leading models in image and signal processing [33, 34], and have been also fruitfully used in many machine learning scenarios such as face recognition [35, 36], abnormal event detection in videos [37], as well as anomaly detection in ECG signals [15, 38]. Domain adaptation techniques for dictionaries have been also widely investigated, mainly to address the image classification tasks [39, 40, 41, 42], and go under the name of *dictionary adaptation*. Among dictionary-adaptation solutions that are mostly related to ours, we mention the framework in [39], which transforms dictionaries while maintaining a domain-invariant sparse representation of the data. Moreover, this framework assumes data in source and target domains share the same sparse representations, which might be inappropriate in case of heartbeats. In [40] dictionary adaptation is performed by learning a sequence of intermediate dictionaries to gradually adapt source to target data, while in [42] a shared discriminative dictionary is learned to provide group-sparse represen-

tations to both source and target domain data. These dictionaries are learned from a training set including both data from source and target domains, which is not possible in ECG monitoring where heartbeats are provided only in resting conditions. The closest alternative to our solution is [41], which performs dictionary adaptation by learning representations for both source and target data in a common low-dimensional subspace. The projections from source and target domains to the common subspace are jointly learned with a dictionary yielding sparse representations of the projected data. In [41] data are transformed by orthogonal projections, while we assume more general linear transformations that can be suitably customized to steer the learning process towards solutions featuring desirable properties.

3. Problem Formulation

We denote by $s\colon \mathbb{N} \to \mathbb{R}$ the ECG signal which has been uniformly sampled in time and preprocessed by standard techniques [13] to remove the baseline wander, i.e. low frequency components due to respiration. We segment the signal by locating the R peaks (see Figure 1) using the Pan-Tompkins algorithm [43] and extract heartbeats $\mathbf{s} \in \mathbb{R}^p$ accordingly:

$$\mathbf{s} = \{s(t+v) \ : \ v \in \mathcal{V}\}, \tag{1}$$

where $\mathcal{V}$ is a neighborhood of the origin containing p samples, and t denotes the index of the R peak in s. Our primary goal is to monitor the ECG signal and detect anomalous heartbeats as they are acquired. To this purpose, we learn a data-driven model that describes the morphology of normal heartbeats and we address the following anomaly detection and domain adaptation problems.

Anomaly detection. We assume that the normal heartbeats of each user u are generated by a user-specific stochastic process $\mathcal{N}_u$. In contrast, anomalous heartbeats are generated by another stochastic process $\mathcal{A} \neq \mathcal{N}_u$ and exhibit a different morphology. Anomalies might be due, for instance, to arrhythmias, acquisition errors or movements that occur during long-term monitoring. In particular, we focus on the detection of anomalies that require analyzing each

heartbeat independently, and we do not consider anomalies that affect, for instance, the heart-rate or that require inspecting multiple heartbeats.

The anomaly-detection problem boils down to defining a decision rule to determine whether an heartbeat conforms or not to the learned model. Our goal is to define a function $e\colon \mathbb{R}^p \to \mathbb{R}$, and a threshold $\gamma_u \in \mathbb{R}$ such that

$$\mathbf{s} \text{ is anomalous} \qquad \Leftrightarrow \qquad e(\mathbf{s}) > \gamma_u \tag{2}$$

To this end we assume a training set containing only normal heartbeats of the user is provided, thus we learn a data-driven model that approximates $\mathcal{N}_u$.

Domain Adaptation. Since heartbeats get transformed when the heart rate changes, the stochastic process generating normal heartbeats actually depends on the heart rate r, thus will be indicated by $\mathcal{N}_{u,r}$. Domain adaptation consists in modifying the anomaly-detection algorithm to correctly operate at different heart rates at a controlled FPR. In our case this requires adapting both the user-specific model and the decision rule.

Our modeling assumption is that normal heartbeats admit a sparse representation w.r.t. to a dictionary $D_{u,r_0} \in \mathbb{R}^{p(r_0)\times n}$, namely

$$\mathbf{s}_{u,r_0} \approx D_{u,r_0}\mathbf{x}_{u,r_0}, \tag{3}$$

where $\mathbf{s}_{u,r_0} \in \mathbb{R}^{p(r_0)}$ denotes an heartbeat acquired at heart rate r_0 (i.e. in the source domain) and its representation $\mathbf{x}_{u,r_0} \in \mathbb{R}^n$ has only few nonzero components. The number of samples in each heartbeat in (3) is $p(r_0)$ which also depends on the heart rate. We formulate dictionary adaptation as the problem of learning, for each target heart rate r, a user-independent transformation $\mathcal{F}_{r,r_0}\colon \mathbb{R}^{p(r_0)\times n} \to \mathbb{R}^{p(r)\times n}$ that operates as follows

$$D_{u,r} = \mathcal{F}_{r,r_0}(D_{u,r_0}) \quad \forall u, \tag{4}$$

and such that $D_{u,r}$ can properly approximate $\mathcal{N}_{u,r}$.

To adapt the decision rule (2) we tackle the problem of learning a set of user-independent transformations $f_{r,r_0}\colon \mathbb{R} \to \mathbb{R}$ to detect anomalies at several heart rates. Considering (2), this consists in transforming the threshold γ_{u,r_0}:

Algorithm 1 An overview of the anomaly-detection algorithm

Input: User-specific dictionary D_{u,r_0}, threshold γ_{u,r_0}, heartbeat $\mathbf{s}_{u,r_0}$.

Output: Label of the heartbeat $\mathbf{s}_{u,r_0}$ (normal or anomalous).

1: Compute the sparse representation $\mathbf{x}_{u,r_0}$ of $\mathbf{s}$ w.r.t. D_{u,r_0} by solving (6).
2: Compute the distance between $\mathbf{s}$ and $\mathfrak{D}_{u,r_0}$ as $e(\mathbf{s}_{u,r_0}) = \|\mathbf{s}_{u,r_0} - D_{u,r_0}\mathbf{x}_{u,r_0}\|_2$.
3: **if** $e(\mathbf{s}_{u,r_0}) > \gamma_{u,r_0}$ **then**
4: Return "Anomalous".
5: **else**
6: Return "Normal".
7: **end if**

$$\gamma_{u,r} = f_{r,r_0}(\gamma_{u,r_0}). \tag{5}$$

The user-independent transformations $\{\mathcal{F}_{r,r_0}\}$ and $\{f_{r,r_0}\}$ have to be learned from a collection $\{S_{u,r}\}$ of sets of normal heartbeats acquired at several heart rates from $L \gg 1$ users. We extract this collection from large publicly available datasets containing long ECG signals.

4. Online Anomaly Detection in ECG Signals

We consider the simple, yet effective, anomaly-detection algorithm [15] presented in Algorithm 1, where normal heartbeats are modeled by means of a user-specific dictionary. The dictionary $D_{u,r_0} \in \mathbb{R}^{p(r_0)\times n}$ and the sparsity level $\kappa \in \mathbb{N}$ define a union of low-dimensional subspaces $\mathfrak{D}_{u,r_0}$, where each subspace is spanned by at most κ columns (i.e., *atoms*) of D_{u,r_0}. We assume that normal heartbeats of user u at heart rate r_0 live close to such union of low-dimensional subspaces. The rationale behind our anomaly detection algorithm is that anomalous heartbeats live far from $\mathfrak{D}_{u,r_0}$.

To compute the distance between $\mathbf{s}$ and $\mathfrak{D}_{u,r_0}$, we have to preliminary compute the projection of $\mathbf{s}$ onto $\mathfrak{D}_{u,r_0}$. To this end, we solve the following *sparse coding* problem [19], obtaining the sparse representation of $\mathbf{s}$ w.r.t. D_{u,r_0}:

$$\mathbf{x}_{u,r_0} = \arg\min_{\mathbf{x}} \|\mathbf{s} - D_{u,r_0}\mathbf{x}\|_2, \ \text{ s.t. } \ \|\mathbf{x}\|_0 \leq \kappa, \tag{6}$$

where $\|\mathbf{x}\|_0$ denotes the number of nonzero components of $\mathbf{x}$. The constraint $\|\mathbf{x}\|_0 \leq \kappa$ promotes the sparsity of $\mathbf{x}$ and bounds the dimension of each subspace

in $\mathfrak{D}_{u,r_0}$ to be at most κ, where $\kappa \ll p(r_0)$. Since (6) is an NP-hard problem [44], we have to resort to greedy algorithms that achieve suboptimal solutions. In particular, we use the Orthogonal Matching Pursuit (OMP) algorithm [45], as this can be substantially accelerated when D_{u,r_0} is underdetermined, i.e. $p(r_0) > n$, as in the case of dictionaries learned for heartbeats. This will be discussed in Section 6.1.

Solution of (6) provides $D_{u,r_0}\mathbf{x}_{u,r_0}$, which is a projection of $\mathbf{s}$ on $\mathfrak{D}_{u,r_0}$. Thus, the distance between $\mathbf{s}_{u,r_0}$ and $\mathfrak{D}_{u,r_0}$ is defined as

$$e(\mathbf{s}_{u,r_0}) = \|\mathbf{s}_{u,r_0} - D_{u,r_0}\mathbf{x}_{u,r_0}\|_2 . \tag{7}$$

According to the decision rule (2), any heartbeat $\mathbf{s}_{u,r_0}$ such that $e(\mathbf{s}_{u,r_0}) > \gamma_{u,r_0}$ is detected as anomalous.

4.1. Dictionary Learning and Threshold Estimation

The user-specific dictionary D_{u,r_0} has to be learned every time that the wearable device is positioned [15]. Let $S_{u,r_0} \in \mathbb{R}^{p(r_0)\times m}$ be a training set of normal heartbeats, stacked column-wise, acquired from user u at heart rate r_0. We learn D_{u,r_0} by solving the following *dictionary learning* problem:

$$D_{u,r_0} = \underset{D\in\mathbb{R}^{p(r_0)\times n},\, X\in\mathbb{R}^{n\times m}}{\arg\min} \|DX - S_{u,r_0}\|_2 , \;\; \text{s.t. } \|\mathbf{x}_i\|_0 \le \kappa, \;\; i = 1,\ldots,n \tag{8}$$

where $\mathbf{x}_i$ denotes the i-th column of the matrix $X \in \mathbb{R}^{n\times m}$ that stacks the coefficient vectors of the heartbeats in S_{u,r_0}. The problem (8) is also NP-Hard and is typically addressed by alternating between the solution w.r.t. D while keeping X fixed (*dictionary update*) and the solution w.r.t. X while keeping D fixed (*sparse coding*). In particular, we adopt K-SVD [18] as this state-of-the-art algorithm has been successfully used to learn models of ECG signals [15, 38].

The threshold γ_{u,r_0} in (2) controls the amount of false positives and is set as the $1-\alpha$ quantile of the empirical distribution of $e(\cdot)$ computed on a training set of normal heartbeats, where α is the desired false positive rate (FPR). To avoid overfitting when defining γ_{u,r_0}, we adopt two disjoints set for learning

D_{u,r_0} and for computing γ_{u,r_0}.

Few minutes of ECG signals acquired in resting conditions are enough to learn the dictionary D_{u,r_0}. A legitimate question is whether anomalous heartbeats in the training set S_{u,r_0} would affect the overall monitoring. In [15] we investigated the impact of heartbeats corrupted by user movements and show that $2\% - 4\%$ of outliers in S_{u,r_0} does not substantially affect the anomaly-detection performance, since these outliers do not feature any specific structure to be learned and matched with heartbeats to be tested. Moreover, in wearable device monitoring heartbeats corrupted by user movements can be removed by analyzing MEMS accelerometers. Unfortunately, D_{u,r_0} might learn arrhythmias that occurs in S_{u,r_0}, as these feature a specific structure. Robust dictionary learning algorithms, such as [46], might mitigate this problem, but the risk of learning as normal these shapes remains. The anomaly-detection algorithm would in this case consider as normal each heartbeat similar to the arrhythmias provided for training. Thus, to monitor patients that are frequently affected by arrhythmias, it is necessary to preliminary screen S_{u,r_0} to remove all the heartbeats that need to be detected during online monitoring. This screening can be performed either by an expert or automatically, e.g., with an offline classifier such as [13, 5] on a remote host, and does not have to be executed during the online monitoring.

5. Domain Adaptation for Online ECG Monitoring

Here we address the domain-adaptation problem, and present a solution to adapt our user-specific anomaly-detection algorithm by means of user-independent transformations that depend only on the source and target heart rates. In particular, we show that these transformations can be successfully learned from datasets containing heartbeats of several users at different heart rates, like [20]. While the anomaly-detection algorithm has to be configured every time the device is positioned, these transformations have to be learned offline and only once, and used during the online monitoring. In the following, we tackle the problem of learning transformations to adapt both the dictionary D_{u,r_0} (Section 5.1),

and threshold γ_{u,r_0} (Section 5.2) to control the false positive rate.

5.1. Dictionary Adaptation

Our goal is to learn a collection of user-independent transformations $\{\mathcal{F}_{r,r_0}\}$ that maps each subspace in $\mathfrak{D}_{r,r_0}$ into a subspace in $\mathfrak{D}_{u,r}$. To preserve the subspace property, we set each $\mathcal{F}_{r,r_0}$ to be a linear function from $\mathbb{R}^{p(r_0)\times n}$ to $\mathbb{R}^{p(r)\times n}$. A linear function between two such vector spaces has in general $p(r_0)p(r)n^2$ degrees of freedom, which is quite a large number when the heartbeats are composed of hundred samples ($p \approx 150$). To reduce the number of parameters, thus the risk of overfitting, we constrain the linear transformation $\mathcal{F}_{r,r_0}$ to have a specific shape that reflects our modeling assumption (3). More precisely, these transformations have to map each atom of D_{u,r_0} to an atom of $D_{u,r}$, thus each generator of the subspaces in $\mathfrak{D}_{u,r_0}$ is mapped to a generator of the subspaces in $\mathfrak{D}_{u,r}$. In this case, $\mathcal{F}_{r,r_0}$ is described by a matrix $F_{r,r_0} \in \mathbb{R}^{p(r)\times p(r_0)}$:

$$D_{u,r} = \mathcal{F}_{r,r_0}(D_{u,r_0}) = F_{r,r_0} D_{u,r_0}, \tag{9}$$

and the number of degrees of freedom of $\mathcal{F}_{r,r_0}$ reduces to $p(r)p(r_0)$ which is the number of entries of the matrix F_{r,r_0}. We point out that the dictionary-adaptation solution in (9) follows from the simple geometrical interpretation of dictionary yielding sparse representation. More complex models of heartbeats might not be straightforwardly adapted to track heart rate variations.

We learn F_{r,r_0} from the datasets in [20] by extracting pairs of training sets $\{S_{u,r_0}\}$ and $\{S_{u,r}\}$ for many users $u \in \{1,\dots,L\}$ and solve the following optimization problem:

$$F_{r,r_0} = \underset{F,\{X_u\}_u}{\arg\min} \frac{1}{2}\sum_{u=1}^{L} \|S_{u,r} - F D_{u,r_0} X_u\|_2^2 + \mu \sum_{u=1}^{L} \|X_u\|_1 + \frac{\lambda}{2}\|W \odot F\|_2^2 + \xi \|W \odot F\|_1 , \tag{10}$$

where the columns of $X_u \in \mathbb{R}^{n\times m}$ contain the sparse w.r.t. FD_{u,r_0} of the corresponding heartbeats in $S_{u,r}$, and the symbol $\odot$ denotes the Hadamard product. All the norms in (10) are vector norms, rather than matrix norms.

In what follows we describe the role of each term in (10): $\|S_{u,r} - FD_{u,r_0}X_u\|_2^2$ assesses how good the transformed dictionary is at approximating the heartbeats

of the user u. This term sums up the reconstruction error over training heartbeats for the L users, and guarantees that dictionaries transformed by F_{r,r_0} can properly describe heartbeats in the target domain. We adopt three regularization terms controlled by the non-negative regularization parameters λ, μ, ξ, and a suitable weight matrix $W \in \mathbb{R}^{p(r)\times p(r_0)}$. The first two regularization terms guarantee that the each transformed dictionary FD_{u,r_0} provides sparse representations to the heartbeats in the target domain, for each user u. We adopt an ℓ^1 regularization to enforce sparsity as a customary choice in the literature [47] since the ℓ^1 norm is convex. The other two terms represent a weighted elastic net penalization over F, which improves the stability of the optimization problem, and the weighting matrix W introduces some a priori information about the transformation $\mathcal{F}_{r,r_0}$ in the minimization. In our case, we expect $\mathcal{F}_{r,r_0}$ to be local, namely that each sample of a transformed atom is determined by only few neighboring samples in the input atom in $D_{u,r}$. Therefore, W features larger weights in positions far from the diagonal of F, and small weights close to the diagonal. In particular, we define the entries of W using Gaussian weights:

$$w_{ij} = 1 - c \cdot e^{-\frac{(j-i)^2}{\sigma}}, \quad i \in \{1, \dots, p(r)\},\ j \in \{1, \dots, p(r_0)\}, \tag{11}$$

where $\sigma > 0$ determines the width of the Gaussian and $c > 0$ is a constant to ensure that $0 \leq w_{ij} \leq 1$, $\forall\, i, j$. An example of W is shown in Figure 2(a).

Solving (10) is not straightforward since it is not jointly convex in X_u and F. However, the functional to be minimized is convex with respect to each variable when the other one is fixed. Therefore, we solve it by the Alternating Direction Method of Multipliers (ADMM) [48], which has been shown to enjoy good convergence properties in these settings [49]. The rationale behind the ADMM is to split the optimization problem in easier sub-problems, and alternate their optimization. To this purpose, we reformulate (10) in an equivalent form:

$$\begin{aligned} &\underset{F,\{X_u\},G,\{Y_u\}}{\arg\min} \frac{1}{2}\sum_{u=1}^{L} \|S_{u,r} - FD_{u,r_0}X_u\|_2^2 + \mu \sum_{u=1}^{L} \|Y_u\|_1 + \frac{\lambda}{2}\|W \odot G\|_2^2 + \xi \|W \odot G\|_1, \\ &\text{s.t.} \qquad F - G = 0, \quad X_u - Y_u = 0,\ \forall u \end{aligned} \tag{12}$$

where $G \in \mathbb{R}^{p(r)\times p(r_0)}$ and $Y_u \in \mathbb{R}^{n\times m}$ are auxiliary variables. According to

the ADMM framework, we define the Augmented Lagrangian [48] of (12) and solve it by alternating the optimization of the following sub-problems:

$$X_u^{(k+1)} = \arg\min_{X_u} \frac{1}{2} \|S_{u,r} - FD_{u,r_0}X_u\|_2^2 + \frac{\rho}{2} \left\| X_u + Y_u^{(k)} + Z_u^{(k)} \right\|_2^2, \tag{13}$$

$$Y_u^{(k+1)} = \arg\min_{Y_u} \mu \|Y_u\|_1 + \frac{\rho}{2} \left\| X_u^{(k+1)} - Y_u + Z_u^{(k)} \right\|_2^2, \tag{14}$$

$$F^{(k+1)} = \arg\min_{F} \frac{1}{2} \sum_{u=1}^{L} \left\| S_{u,r} - FD_{u,r_0}X_u^{(k+1)} \right\|_2^2 + \frac{\rho}{2} \left\| F + G^{(k)} + H^{(j)} \right\|_2^2, \tag{15}$$

$$G^{(k+1)} = \arg\min_{G} \frac{\lambda}{2} \|W \odot G\|_2^2 + \xi \|W \odot G\|_1 + \frac{\rho}{2} \left\| F^{(k+1)} - G + H_k \right\|_2^2, \tag{16}$$

$$Z_u^{(k+1)} = Z_u^{(k)} + X_u^{(k+1)} - Y_u^{(k+1)}, \tag{17}$$

$$H^{(k+1)} = H^{(k)} + F^{(k+1)} - G^{(k+1)}, \tag{18}$$

where $Z_u \in \mathbb{R}^{n \times m}$ and $H \in \mathbb{R}^{p(r) \times p(r_0)}$ are the scaled Lagrange multipliers of the constraints in (12), that are updated in (17) and (18), respectively. Sub-problems (13) and (15) are quadratic expressions which can be efficiently solved by Gaussian elimination. Problem (14) admits a closed-form solution which corresponds to the proximal mapping [50] of the function $(\mu/\rho)|| \cdot ||_1$

$$\left[Y_u^{(k+1)} \right]_{ij} = \mathcal{S}_{\mu/\rho} \left(\left[X_u^{(k+1)} + Z_u^{(k)} \right]_{ij} \right), \tag{19}$$

where $[\cdot]_{ij}$ denotes a matrix entry, and $\mathcal{S}_\eta \colon \mathbb{R} \to \mathbb{R}$:

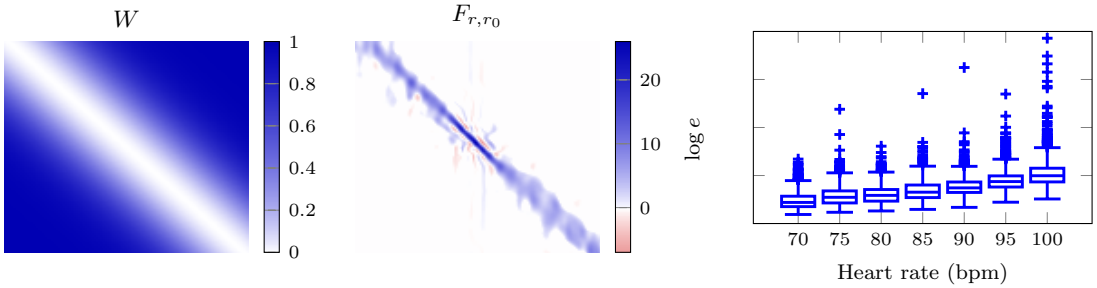
$$\mathcal{S}_\eta(x) = \text{sign}(x) \cdot \max\{0, x - \eta\}, \tag{20}$$

is the soft-thresholding operator. From Theorem 4 in [50] and simple algebra, it follows that also (16) can be solved by soft thresholding:

$$\left[G^{(k+1)} \right]_{ij} = \frac{1}{1 + \lambda w_{ij}^2} \mathcal{S}_{\xi w_{ij}/\rho} \left(\left[F^{(k+1)} + H^{(k)} \right]_{ij} \right). \tag{21}$$

To initialize the ADMM algorithm, we set to zero the values of all the variable but $F^{(0)}$, which is initialized to uniformly distributed random values to avoid trivial solutions. Then, we iteratively solve (13–18) until a maximum number of iterations is reached or the primal and dual residuals [48] fall below given thresholds.

Figure 2(b) shows an example of learned F_{r,r_0}, which, as expected, is local, since the nonzero elements of F_{r,r_0} are concentrated at the diagonal.



(a) (b) (c)

Figure 2: (a) The weight matrix W used in (10) to learn the matrix F_{r,r_0}. The elements of W are set according to (11) to enforce the learned transformation to be local: the elements on the diagonal are small, while the ones far from the diagonal are large. (b) Example of learned F_{r,r_0} for $r = 90$, $r_0 = 70$ using the weight matrix W shown in (a). The learned transformation is local, since nonzero elements of F_{r,r_0} are concentrated around the diagonal. (c) Boxplot of the $\log e(\mathbf{s}_{u,r})$ computed on normal heartbeats acquired at different the heart rate r. In both cases the distribution of $\log e(\mathbf{s}_{u,r})$ seems to shift of a value that increases linearly with the heart rate r. This support pour choice of transformation in (22).

The same optimization problem is solved for several pairs (r, r_0) to learn the collection $\{\mathcal{F}_{r,r_0}\}$ that is used to adapt the user-specific dictionary D_{u,r_0} to any target heart rate r during the online monitoring.

5.2. Threshold Adaptation

Adapting the dictionary D_{u,r_0} when the heart rate changes is not enough to enable long-term ECG monitoring. In fact, also the threshold γ_{u,r_0} in the decision rule (2) has to be adapted. To this purpose, we learn a set $\{f_{r,r_0}\}$ of user-independent transformations by solving the following optimization problem:

$$\gamma_{u,r} = f_{r,r_0}(\gamma_{u,r_0}) = \gamma_{u,r_0} \cdot \exp(a(r - r_0)), \tag{22}$$

where $a \in \mathbb{R}$ is the only parameter that has to be learned.

The choice of transformation in (22) is justified by an empirical consideration. Boxplots from Figure 2(c) report the empirical distribution of $\log e_{u,r}$ for two different users at heart rates r, being $e_{u,r}$ the random variable corresponding to $e(\mathbf{s}_{u,r})$ when $\mathbf{s}_{u,r}$ is drawn from the stochastic process $\mathcal{N}_{u,r}$ and the sparse coding is performed w.r.t. the adapted dictionary $D_{u,r}$. The trend of these boxplots suggests that $(\log e_{u,r} - \log e_{u,r_0})$ is proportional to $(r - r_0)$. This observation implies the following relation:

$$\log e_{u,r} = \log e_{u,r_0} + a(r - r_0), \tag{23}$$

where $a > 0 \in \mathbb{R}$ is the parameter yielding the first order approximation to the trend of such distributions. To maintain a fixed false positive rate (FPR) when the heart rate changes, we need to set $\gamma_{u,r}$ to guarantee a constant probability of considering a normal heartbeat $\mathbf{s}_{u,r}$ as anomalous:

$$P(e_{u,r} > \gamma_{u,r}) = P(e_{u,r_0} > \gamma_{u,r_0}). \tag{24}$$

Combining (23) and (24), we can derive the relation between γ_{u,r_0} and $\gamma_{u,r}$:

$$\begin{aligned} P\left(e_{u,r_0} > \gamma_{u,r_0}\right) &= P\left(e_{u,r} > \gamma_{u,r}\right) = P\left(\log e_{u,r} > \log \gamma_{u,r}\right) = \\ &= P\left(\log e_{u,r_0} + a(r - r_0) > \log \gamma_{u,r}\right) = \\ &= P\left(e_{u,r_0} > \gamma_{u,r} \exp(-a(r - r_0))\right). \end{aligned} \tag{25}$$

The last equation implies that $\gamma_{u,r_0} = \gamma_{u,r} \exp(-a_{r,r_0}(r - r_0))$, thus (22).

We now address the problem of estimating such transformations from a training set containing heartbeats of multiple users. Figure 2(c) suggests that the distribution of $\log e_{u,r}$ is symmetric, thus we write

$$\log e_{u,r} = b_{u,r} + \eta_u, \tag{26}$$

where $b_{u,r}$ is the expected value of $\log e_{u,r}$, while η_u is a stochastic term that has a symmetric distribution w.r.t the origin. Taking the expected value in (23), we have that $b_{u,r} = b_{u,r_0} + a(r - r_0)$, which we can substitute in (26) and obtain:

$$\log e_{u,r} = b_{u,r_0} + a \cdot (r - r_0) + \eta_u, \tag{27}$$

which is a linear regression model for $\log e_{u,r}$ w.r.t. the heart rate r. The parameter b_{u,r_0} is user-dependent and can be estimated from S_{u,r_0}, while we can estimate a via least squares over multiple users by solving

$$a = \arg\min_{\tilde{a}} \frac{1}{2} \sum_{u=1}^{L} \sum_{j=1}^{R} \sum_{i=1}^{m} \left(\log\left(e(\mathbf{s}_{u,r_j}(i))\right) - b_{u,r_0} - \tilde{a} \cdot (r_j - r_0)\right)^2, \tag{28}$$

where b_{u,r_0}, $u \in \{1, \ldots, L\}$ is the average value of $e(\mathbf{s}_{u,r_0})$ over $\{S_{u,r_0}\}$. We estimate a by setting to 0 the derivative of the functional in (28):

$$a = \frac{\sum_{u=1}^{L} \sum_{j=1}^{R} \sum_{i=1}^{m} \left(\log\left(e(\mathbf{s}_{u,r_j}(i))\right) - b_u\right) \cdot (r_j - r_0)}{mL \sum_{j=1}^{R} (r_j - r_0)^2}. \tag{29}$$

The estimated a defines the user-independent transformation in (22), to adapt the decision rule. Our anomaly-detection algorithm can thus perform long-term ECG monitoring by transforming the user specific dictionary D_{u,r_0} and the threshold γ_{r_0} by means of the transformations in (9) and (22), respectively.

6. Wearable Device Monitoring

In this section we describe how our solution can be implemented on a low power wearabale device to perform online and long-term ECG monitoring. We consider the Bio2Bit-Dongle [21], a wearable device developed by STMicroelectronics that is composed by the Bio2Bit Move [51] and a small *dongle*. The Bio2Bit Move sensing device is plugged on a flexible chest strap and transmits the ECG signal via Bluetooth Low Energy (BLE) to the dongle, that analyzes in real time the received signal. In particular, the dongle segments each heartbeat $\mathbf{s}$, estimates the heart rate and determines whether $\mathbf{s}$ is normal or anomalous using Algorithm 1. To this end, we present an efficient variant of the OMP algorithm [45] that is specifically designed for undercomplete dictionaries (Section 6.1); the whole monitoring scheme is in Section 6.2, while Section 6.3 summarizes the configuration procedures to be performed ever time device is placed.

6.1. Optimized OMP for Undercomplete Dictionaries

Computational costs are a critical aspect for wearable devices: since the sparse-coding has to be solved for each acquired heartbeat, a reduction in the number of calculations performed can meaningfully extend the battery life of the device. Therefore, here we propose an alternative solution of the sparse coding problem (6), that in case of undercomplete dictionaries is substantially lighter than traditional OMP. To simplify notation, we omit (u, r) from the subscript indexes, since all the variables refer to a user u and a target heart-rate r.

In our previous study [15] we showed that in ECG monitoring the anomaly-detection algorithm achieves the best performance when the dictionary D is undercomplete, namely the number of atoms n is smaller than the dimension p of the heartbeats. In these cases, D does not span the entire space $\mathbb{R}^p$, but only

a subspace $\mathcal{U}$ having dimension n that includes the union of low dimensional subspaces $\mathfrak{D}$ identified by D. Since solving the sparse coding (6) corresponds to computing the projection of $\mathbf{s}$ onto $\mathfrak{D}$, we can reduce the number of operations performed in the OMP by first projecting $\mathbf{s}$ onto $\mathcal{U}$, and then solving the sparse coding problem on the projected heartbeat. To this end, we select an orthonormal basis of $\mathcal{U}$ by computing the QR decomposition of $D = QR$, being $R \in \mathbb{R}^{n\times n}$ an upper-triangular matrix, and $Q \in \mathbb{R}^{p\times n}$ such that $Q^TQ = I_n$, where I_n is the $n \times n$ identity matrix. Then, we address the following sparse-coding problem through OMP instead of (6)

$$\widehat{\mathbf{x}} = \arg\min_x \left\|Q^T\mathbf{s} - R\mathbf{x}\right\|_2^2, \text{ s.t. } \|\mathbf{x}\|_0 \leq \kappa, \tag{30}$$

where $Q^T\mathbf{s}$ is the projection of $\mathbf{s}$ onto $\mathcal{U}$.

In what follows, we prove that problems (6) and (30) have the same solution. However, since (30) is much cheaper than (6) since the computational complexity of OMP is determined by the target sparsity κ and the number of atoms n. Solving (6) with a dictionary $D \in \mathbb{R}^{p\times n}$ yields a complexity $O(\kappa pn)$ [52]. In contrast, since $R \in \mathbb{R}^{n\times n}$, solving (30) requires yields a complexity $O(\kappa n^2)$, thus the computational complexity of OMP is dominated by the cost of the matrix product $Q^T\mathbf{s}$, i.e. $O(pn)$. In practice, we reduce the overall cost of the OMP algorithm by a factor κ, from $O(\kappa pn)$ to $O(pn)$. The following proves the equivalence of (6) and (30).

Proposition 1. *Let $Q \in \mathbb{R}^{p\times n}$ and $R \in \mathbb{R}^{n\times n}$ define the QR decomposition of the dictionary $D \in \mathbb{R}^{p\times n}$. Then, for every $\mathbf{s} \in \mathbb{R}^p$ and $\mathbf{x} \in \mathbb{R}^n$ it holds:*

$$\|\mathbf{s} - D\mathbf{x}\|_2^2 = \left\|Q^T\mathbf{s} - R\mathbf{x}\right\|_2^2 + \|\mathbf{s}\|_2^2 - \left\|Q^T\mathbf{s}\right\|_2^2. \tag{31}$$

Proof. Let us first remark that the columns of Q form an orthonormal basis of the n-dimensional subspace $\mathcal{U} \subset \mathbb{R}^p$ spanned by the columns of D. Since $n < p$, it is always possible to define a set of $(p - n)$ orthonormal vectors to extend Q to an orthonormal basis of the entire $\mathbb{R}^p$. In particular, we define $P = [Q, Q_\perp]$, where $Q_\perp \in \mathbb{R}^{p\times n-p}$ is such that $Q_\perp^TQ = 0$ and $Q_\perp^TQ_\perp = I_{p-n}$. Since P is an orthogonal matrix, for each $\mathbf{v} \in \mathbb{R}^p$ we have that:

Algorithm 2 Optimized Online ECG Monitoring

Input: Heartbeat $\mathbf{s}$, associated heart rate $\bar{r}$, user-specific matrices $\{Q_{u,r}\}_r$ and $\{R_{u,r}\}_r$ and thresholds $\{\gamma_{u,r}\}_r$ provided by the user configuration.

Output: Label of the heartbeat $\mathbf{s}$ (normal or anomalous).

1: Select the matrices $Q_{u,r}, R_{u,r}$ and the threshold $\gamma_{u,r}$ that are associated to the current heart rate $\bar{r}$.
2: Project $\mathbf{s}$ onto the subspace $\mathcal{U}$ containing $\mathfrak{D}_{u,r}$ by computing $Q_{u,r}^T\mathbf{s}$.
3: Compute the sparse representation $\mathbf{x}_{u,r}$ of $\mathbf{s}$ w.r.t. $D_{u,r} = Q_{u,r}R_{u,r}$ by (30).
4: Compute the distance $e(\mathbf{s})$ between $\mathbf{s}$ and $\mathfrak{D}_{u,r}$ as in (34)
5: **if** $e(\mathbf{s}) > \gamma_{u,r}$ **then**
6: Return "Anomalous".
7: **else**
8: Return "Normal".
9: **end if**

$$\|\mathbf{v}\|_2^2 = \left\|P^T\mathbf{v}\right\|_2^2 = \left\|Q^T\mathbf{v}\right\|_2^2 + \left\|Q_\perp^T\mathbf{v}\right\|_2^2. \tag{32}$$

Substituting $D = QR$ in $\|\mathbf{s} - D\mathbf{x}\|_2^2$ and applying (32) to $\mathbf{v} = \mathbf{s} - D\mathbf{x}$ we obtain:

$$\begin{aligned}\|\mathbf{s} - D\mathbf{x}\|_2^2 = \|\mathbf{s} - QR\mathbf{x}\|_2^2 = \left\|P^T(\mathbf{s} - QR\mathbf{x})\right\|_2^2 = \\ = \left\|Q^T\mathbf{s} - Q^TQR\mathbf{x}\right\|_2^2 + \left\|Q_\perp^T\mathbf{s} - Q_\perp^TQR\mathbf{x}\right\|_2^2 = \\ = \left\|Q^T\mathbf{s} - R\mathbf{x}\right\|_2^2 + \left\|Q_\perp^T\mathbf{s}\right\|_2^2,\end{aligned} \tag{33}$$

where the last equality holds since $Q_\perp^TQ = 0$ and $Q_\perp^TQ_\perp = I_{p-n}$. Proposition is proven by substituting $\mathbf{v} = \mathbf{s}$ in (32), yielding $\left\|Q_\perp^T\mathbf{s}\right\|_2^2 = \|\mathbf{s}\|_2^2 - \left\|Q^T\mathbf{s}\right\|_2^2$, and replacing this expression in (33). □

Proposition 1 confirms that we can substitute the functional in (6) with the right-hand side of (31). The last two terms of (31) do not depend on $\mathbf{x}$ and can be ignored in the minimization, thus the problems (6) and (30) are equivalent.

6.2. Online Monitoring on Wearable Devices

Algorithm 2 details the steps to be performed during the user-configuration phase described in Section 6.3. To reduce the computational complexity, we pre-compute and store on the device all the matrices $\{Q_{r,r_0}\}$ and $\{R_{r,r_0}\}$ and thresholds. Each heartbeat $\mathbf{s}$ to be tested is accompanied by its heart rate $\bar{r}$ that identifies which matrices $Q_{u,r}$ and $R_{u,r}$, and threshold $\gamma_{u,r}$ to use for

detecting anomalies (Algorithm 2 line 1). The heartbeat $\mathbf{s}$ is then projected in the subspace generated by columns of $Q_{u,r}$, by computing $Q_{u,r}^T\mathbf{s}$ (line 2), and the OMP algorithm is executed using the signal $Q_{u,r}^T\mathbf{s}$ and the dictionary $R_{u,r}$ (line 3). Finally, the distance $e(\mathbf{s})$ between the heartbeat $\mathbf{s}$ and the union of low dimensional subspaces $\mathfrak{D}$ is computed (line 4) as

$$e(\mathbf{s}) = \|\mathbf{s} - D\mathbf{x}\|_2 = \sqrt{\|Q^T\mathbf{s} - R\mathbf{x}\|_2^2 + \|\mathbf{s}\|_2^2 - \|Q^T\mathbf{s}\|_2^2}, \tag{34}$$

and compared with $\gamma_{u,r}$ to determine whether $\mathbf{s}$ is anomalous or not (line 5).

We measure the execution time of the standard OMP and our variant on the development board NUCLEO-L476RG [53], which embeds the same ultra-low power microcontroller unit embedded on the Dongle, and that has been equipped by a BLE evaluation board to communicate with the Bio2Bit Move. The average time required to compute $e(\mathbf{s})$ using our optimized variant of the OMP is 1.360 ms which is about 48% less than the time required when using a standard implementation of the OMP algorithm. Moreover, the average current absorbed by the dongle in 20 minutes of ECG monitoring is 10.01 mA, that ensures about 16 hours of monitoring autonomy with a 592 mWh battery, the same of the Bio2Bit Move.

6.3. User Configuration

Algorithm 3 describes the user-configuration phase, which has to be performed every time the Bio2Bit Move is positioned. User configuration includes learning D_{u,r_0} and estimating γ_{u,r_0}, from the training set S_{u,r_0}, as described in Section 4.1. The configuration of the Bio2Bit dongle to perform domain adaptation (Section 5) and execute the optimized OMP (Section 6.1) are described in what follows.

We experienced that 10 minutes of ECG signals acquired in resting conditions are typically enough for the user configuration: acquired signals are initially segmented by detecting R peaks by means of the Pan-Tompkins algorithm [43]. The heart rate $\bar{r}$ associated to each heartbeat is the median of the inverse of distances between two consecutive R peaks over the last 10 seconds,

Algorithm 3 User Configuration

Input: Training set S_{u,r_0}, source heart rate r_0, user-independent transformations $\{\mathcal{F}_{r,r_0}\}$, $\{f_{r,r_0}\}$, desired false positive rate α.

Output: Matrices $\{Q_{u,r}\}$, $\{R_{u,r}\}$, thresholds $\{\gamma_{u,r}\}$.

1: Split the training set S_{u,r_0} in two sets T and V.
2: Learn the user specific dictionary D_{u,r_0} by solving problem (8), using T in place of S_{u,r_0} as training set.
3: Compute $e(\mathbf{s}_{u,r_0})$ for each heartbeat $\mathbf{s}_{u,r_0}$ in V.
4: Set γ_{u,r_0} as the $(1-\alpha)$ quantile of the empirical distribution of $e(\cdot)$ over V.
5: **for** each r in the range $[70, 120]$ **do**
6: Adapt the dictionary D_{u,r_0} by computing $D_{u,r} = \mathcal{F}_{r,r_0}(D_{u,r_0})$.
7: Compute the QR decomposition of the adapted dictionary: $D_{u,r} = Q_{u,r}R_{u,r}$.
8: Adapt the threshold γ_{u,r_0} by computing $\gamma_{u,r} = f_{r,r_0}(\gamma_{u,r_0})$.
9: **end for**

quantized to a resolution of 10 bpm (beats per minute). The most frequent heart rate in these 10 minutes is selected as r_0, and only the heartbeats associated to r_0 belong to the training set S_{u,r_0}. The next operations to configure the device are more computationally demanding than the online monitoring, and can be conveniently performed on a host, such as a smartphone.

The host receives the training set S_{u,r_0} and r_0, and divides S_{u,r_0} into a set used to learn D_{u,r_0} and a set to estimate γ_{u,r_0}, as described in Section 4.1. The host pre-computes $D_{u,r} = \mathcal{F}_{r,r_0}(D_{u,r_0})$ and $\gamma_{u,r} = f_{r,r_0}(\gamma_{r_0})$ for a range of admissible heart rates r (e.g., $[70, 120]$), as well as the QR decomposition of each transformed dictionary $\{D_{u,r}\}_r$. The matrices $Q_{u,r}$ and $R_{u,r}$ are then sent back to the dongle that stores them for the optimized sparse coding (Algorithm 2).

7. Experiments

To validate our solution we perform an extensive experimental campaign, that is meant to show that: *i*) dictionaries yielding sparse representations can be successfully used to detect heartbeats featuring an anomalous morphology, including potentially dangerous arrhythmias (Section 7.4); *ii*) learned user-independent transformations can successfully adapt our anomaly-detection algorithm (Section 7.5); *iii*) our solution can be effectively used to monitor ECG signals acquired from a wearable device (Section 7.6). We describe the datasets

Table 1: Datasets from Physionet [20] used in the experiments

Dataset	Number of users	Average duration (h)	Heart rate resolution (bpm)
LTSTDB	80	22	5
LTAFDB	84	24	5
LTDB	7	16	5
EDB	45	2	10
MIT-BIH	48	0.5	–

used in our experiments in Section 7.1, the figures of merit in Section 7.2 and the alternative solutions in Section 7.3.

7.1. Datasets Description

We consider 5 datasets publicly available from Physionet [20] which have been acquired using Holter devices, and a dataset containing ECG signals recorded using the Bio2Bit Dongle. All the dataset from Physionet have been manually annotated using an automatic tool and corrected by cardiologists. Table 1 reports the number of users in each dataset and the durations of the ECG signals recorded. We use different resolution for heart rate quantization (see Table 1) on the duration of the ECG signal, to guarantee a sufficient number of heartbeats for each quantized heart rate. We consider as anomalous heartbeats all the annotated arrhythmias. The MIT-BIH Arrhythmia dataset contains only short ECG signals and do not present heart rate variability. Therefore, we use this dataset to assess the anomaly-detection performance, while not the domain adaptation performance.

Finally, the *Bio2Bit Move Dataset* (B2B) contains 15 ECG signals recorded by the Bio2Bit Move device. 12 ECG signals are from healthy users, while 3 are from patients affected by a cardiovascular disease and has been annotated by a cardiologist. Each ECG signal in the dataset lasts at least 1 hour and is acquired following a protocol including normal-life activities (e.g. resting, lying down, walking, resting after a small effort), thus the heart-rate significantly varies in each ECG signal. We restrict our analysis to the most frequent heart rates, i.e. $\{70, 80, 90, 100\}$. Due to the limited number of leads and their reduced distance, heartbeats in the B2B dataset are very different from those in the datasets from

Physionet. Low-quality heartbeats have been discarded by an automatic tool and the supervision of a cardiologist.

7.2. Figures of Merit

To assess how good the adapted dictionaries are at modeling the heartbeats, we select as figure of merit the distance e of the heartbeats from the transformed union of low dimensional subspaces, that can be also interpreted as the reconstruction error yielded by the sparse approximation (3).

Figures of merit typically used to assess anomaly-detection performance are the True Positive Rate (TPR) and the False Positive Rate (FPR), namely the percentage of anomalous heartbeats that have been detected and the percentage of normal heartbeats identified as anomalous, respectively. Since both FPR and TPR depend on the threshold in (2), we analyze the Receiving Operating Characteristic (ROC) curve by plotting TPR against FPR for different values of the threshold and, as a global indicator of the anomaly-detection performance, the Area Under the ROC Curve (AUC).

7.3. Alternative Solutions

As alternative anomaly-detection algorithm we consider [38], denoted as **Coding**. This algorithm employs dictionaries yielding sparse representations and embeds the anomaly-detection phase in a ad-hoc sparse coding procedure. More precisely, the sparse coding formulation includes an additional term $\mathbf{a}$ which gathers heartbeats that cannot be sparsely represented by D. Anomalies are detected controlling whether the magnitude of $\mathbf{a}$ exceeds a fixed threshold. To enable a fair comparison with our anomaly detector, the two solutions use the same dictionary and have been manually configured to achieve their best performance.

We consider the following dictionary-adaptation solutions:

Cut: since the support of each heartbeat contracts as the heart-rate increases (see Figure 1), the simplest form of adaptation consists in removing the first and the last samples of each column of D_{u,r_0}, thus "cutting" the support of each

atom in the dictionary. This baseline solution transforms each dictionary D_{u,r_0} using a pre-defined transformation which does not require training data.

DTW: this solution is based on dynamic time-warping [54], an established signal-processing algorithm to align two vectors and measure their similarity. Given $\mathbf{v}_0 \in \mathbb{R}^{p_0}$ and $\mathbf{v}_1 \in \mathbb{R}^{p_1}$ where $p_0 \neq p_1$, dynamic time-warping performs a non uniform resampling of $\mathbf{v}_0$ and $\mathbf{v}_1$ to obtain two aligned vectors $\widetilde{\mathbf{v}}_0, \widetilde{\mathbf{v}}_1 \in \mathbb{R}^{p_a}$ that have a common support p_a, such that $p_a \geq p_0$ and $p_a \geq p_1$. The resampling patterns are estimated by minimizing the Euclidean distance between $\widetilde{\mathbf{v}}_0$ and $\widetilde{\mathbf{v}}_1$ through dynamic programming. Since resampling is a linear operation, it is always possible to express DTW by two matrices $A_i \in \mathbb{R}^{p_a \times p_i}$, $i = 0, 1$, such that $\widetilde{\mathbf{v}}_i = A_i \mathbf{v}_i$. The matrices define the transformation for each pair of heartbeats. In particular, we compute A_{r_0} and A_r by aligning the first principal components of S_{u,r_0} and $S_{u,r}$. To obtain user-independent transformations, the resampling patterns are computed to minimize the sum of the Euclidean distance between the aligned first principal components of S_{u,r_0} and $S_{u,r}$ over many users u. The corresponding transformation $\mathcal{F}_{r,r_0}$ to map dictionaries is linear as in (9) and is defined by $F_{r,r_0} = A_r^+ A_{r_0}$, begin A_r^+ the pseudo-inverse of A_r.

SDDL: Shared Domain-adaptive Dictionary Learning is a domain-adaptation algorithm specifically designed for dictionaries [41]. For each source-target domain pairs, SDDL jointly learns two projections from these domains into a common subspace, as well as a shared dictionary providing sparse representations of the projected data. While this solution is claimed to be general in [41], it has been primarily developed for classification purposes, as it learns class-wise mutually incoherent dictionaries. We adapt [41] by learning the projection from the LTSTDB and LTAFDB datasets, and the dictionaries in the shared subspace by means of the KSVD [18] in place of the discriminative dictionary-learning algorithm in [41]. Thus, during the user-specific configuration, we project S_{u,r_0} onto the low-dimensional subspace using the learned projection, then we learn a user-specific (but heart rate independent) dictionary D_u. During online monitoring, we project each heartbeat onto the subspace, then we compute the sparse representation w.r.t. D_u and back-project the obtained reconstruction. The ℓ^2

norm of the difference between the original and the back-projected heartbeats yield the reconstruction error used for anomaly detection.

Oracle: this ideal solution directly learns a dictionary $D_{u,r}$ from a training set $S_{u,r}$ using KSVD in the target domain.As such, this cannot be pursued in practical monitoring, but represents a performance reference for domain-adaptation algorithms.

These domain-adaptation solutions have been trained on all ECG signals in LTSTDB and LTAFDB datasets, which present a large variability both in term of users and heart rates, and tested on all other datasets (Physionet and B2B). Moreover, to assess the performance on all Physionet datasets we learn a set of transformations from LTSTDB dataset and test it on LTAFDB dataset, and viceversa. Finally, since all the domain-adaptation solutions depend on several hyper-parameters, we adopt a 5-fold cross-validation procedure during the training phase, and use random search [55] to select the hyper-parameters that achieve the best heartbeats reconstruction performance on a validation set.

7.4. Arrhythmias Detection Experiments

We assess the anomaly detection performance of our algorithm on the arrhythmias of the MIT-BIH dataset. For each user in the dataset, we compute the ROC curves and the AUC values obtained using our algorithm and Coding algorithm [38]. The coding algorithm achieves a median AUC equal to 0.9935 and outperforms our algorithm (median AUC equal to 0.992), as confirmed by a two-samples sign rank test performed over the populations of AUC values (p-value $= 0.002$). However, our algorithm achieves a very good performance with a computational complexity of $O(\kappa pn)$, that is significantly lower than Coding algorithm, which is $O(n^3t + pnt)$ [38], where t is the number of iterations required by the algorithm to achieve convergence, that is typically several tens.

7.5. Domain Adaptation Experiments

We perform three domain adaptation experiments: the first two are on Physionet datasets, the third one on B2B dataset. The first experiment aims to assess the normal heartbeats reconstruction when adapting dictionaries to track heart

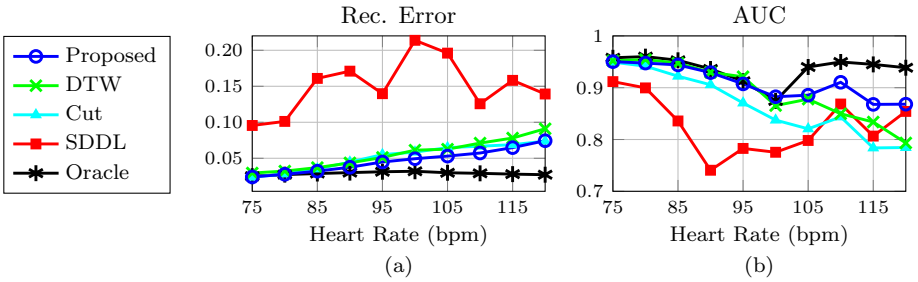


Figure 3: Results of dictionary adaptation experiments on Physionet datasets. (a) The median reconstruction error over all users. As expected, in the Oracle solution the reconstruction error is nearly constant w.r.t. the heart rate r, and increases with r in all domain adaptation solutions, confirming that the change in the heartbeat morphology becomes more evident for large heart rates. Among such solutions, the Proposed one achieves the lowest reconstruction error. (b) The median AUC computed over all the users. As in case of the reconstruction error, the Proposed solution leads to the best performance in case of domain adaptation, although the DTW achieves similar AUC values.

rate changes. More precisely, for each test user u we learn a dictionary D_{u,r_0} from heartbeats at heart rate $r_0 = 70$ bpm and then we analyze the reconstruction error on normal heartbeats w.r.t. the dictionary obtained by adapting D_{u,r_0} on different heart rates. Figure 3(a) shows the median reconstruction error over all the test users and heart rates (the lower the better). Our dictionary adaptation solution outperforms all the alternatives (except for the Oracle), in particular at the high heart rates. A signed-rank test confirms that our solution achieves lower reconstruction error than both DTW and CUT (p-value < 0.00001), except for low heart-rates ($r = 75, 80$), where Proposed and Cut solutions achieve similar performance.

In the second experiment we assess the anomaly-detection performance when the algorithm is adapted to operate at different heart rates on each specific user. Figure 3(b) reports the median AUC computed over all test user for each considered solution. The Proposed one achieves similar performance of DTW for low heart rates, and is the best when $r \geq 100$, although a signed rank test does not report enough statical evidence that the two solutions perform differently. This is not in conflict with results on the reconstruction error. Some anomalous heartbeats can be better perceived at low heart rates and be detected even when the reconstruction w.r.t. the adapted dictionaries is slightly worse.

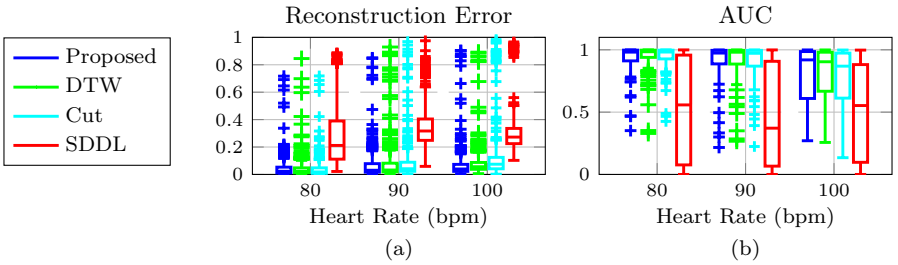


Figure 4: Results of dictionary adaptation on B2B dataset. (a) Boxplots of the reconstruction error computed over all the user in the datasets. (b) Boxplots of the AUC achieved by all solutions to detect *inter-user* anomalous heartbeats. The proposed solution achieves the best performance according to both figures of merit, especially in case of $r = 100$.

Finally, in the third experiment we simulate an online monitoring scenario, where we set a desired FPR $\alpha = 0.01$ and assess the TPR achieved when the D_{u,r_0} and γ_{u,r_0} using the learned transformations. Figure 5(a) shows the median FPR and TPR computed over all the users. The learned transformation successfully maintain the FPR below the target value α for each heart rate. The TPR is large for small heart rates, but decreases when the heart rate becomes significantly larger than r_0. This is probably due to a quality degradation of heartbeats because of user movements which affects the capability of the detector to distinguish between normal and anomalous heartbeats.

7.6. Online Monitoring on Wearable Devices Experiments

Here we show that the transformations learned from Physionet datasets can successfully adapt dictionaries learned from other devices, such as Bio2Bit Dongle, which is completely different from the Holter used to acquire the ECG signals in the Physionet datasets. This is necessary for our solution to enable long-term monitoring. Therefore, we repeat the experiments in Section 7.5 using the same transformations on the B2B dataset to assess both heartbeat reconstruction and anomaly detection performance. For each user u we learn D_{u,r_0} and γ_{u,r_0} from the first 10 minutes of ECG signal, and perform domain adaptation to heart rate $r \in \{80, 90, 100\}$.

Figure 4(a) shows the boxplots of the reconstruction error computed on normal heartbeats of each user for different r. This plot show that the median

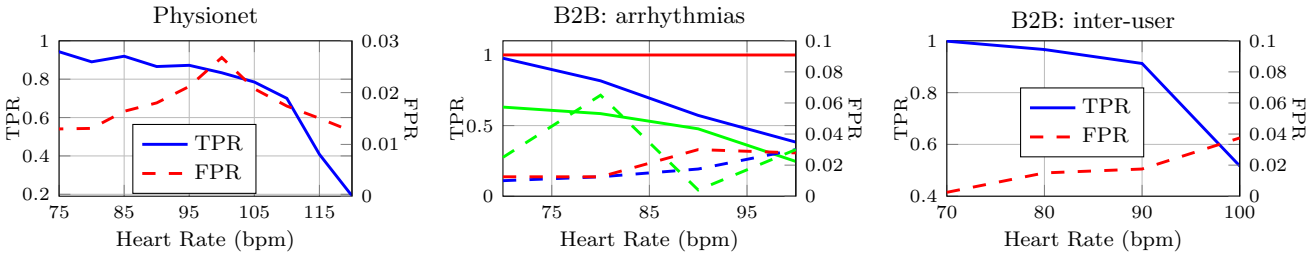


Figure 5: Results of online monitoring experiments, where the threshold γ_{u,r_0} has been set to achieve a desired FPR $\alpha = 0.01$. (a) Median TPR and FPR computed over all the users in Physionet datasets. (b) FPR and TPR computed on the 3 patients affected by cardiovascular diseases in the B2B dataset. Different colors correspond to different users. (c) Median FPR and TPR in the inter-user anomaly detection experiments on the B2B dataset.

reconstruction error of the Proposed solution is lower the others even for large heart rates. As in the experiments on the Physionet datasets, we assess the performance in the online monitoring scenario by setting a desired FPR $\alpha = 0.01$, and computing the actual FPR and TPR score for each heart rate r. Figure 5(b) shows the performance of our solution on the 3 patients affected by cardiovascular diseases: the FPR exceeds the desired value of $\alpha = 0.01$, but it is maintained constant. The TPR is very large for one patient, but it is smaller in case of the other two, where the number of arrhythmias is low, thus the estimate of the TPR is subject to a large variance.

To increase the number of anomalous heartbeats, we pursue the approach described in [15] and artificially introduce *inter-user* anomalies. More precisely, for each healthy user u we consider as anomalous any heartbeat from a different healthy user, which indeed features a morphology that is different from training ones. Figure 4(b) shows the boxplot of the AUC: again, the proposed solution outperforms the others, although DTW performs comparably, as confirmed by a signed rank test (p-value ≈ 0.6). Finally, we assess the anomaly detection performance in the online monitoring scenario also on the inter-user anomalies. Figure 5(c) shows the median FPR and TPR, confirming that our solution successfully detects anomalous heartbeats when the heart rate increases while maintaining constant FPR.

8. Conclusions

We have addressed the challenge of performing online and long-term ECG monitoring directly on a low-power, wearable device. In particular, we learn a dictionary yielding sparse representations of the normal heartbeats of each specific user, and detect as anomalous any heartbeat that do not conform to this dictionary. The sparse coding, which is the most computationally demanding step during monitoring, has been reformulated and optimized to analyze heartbeats in an online manner on a wearable device.

Dictionaries are very interpretable models and we have shown they can be successfully adapted to monitor ECG signals even when the heart rate – thus the heartbeat morphology – changes. Perhaps surprisingly, while dictionaries used to detect anomalies have to be user-specific, they can be successfully adapted by user-independent transformations, learned from large and publicly available datasets. Thus, a few minutes of ECG signals acquired in resting conditions are enough to configure the device for long-term monitoring. Our algorithms have been implemented and successfully tested in a demo device [21] performing online ECG monitoring.

The proposed solution is still sensitive to user movements, which might corrupt the heartbeat morphology and result in false alarms. This can prevent the ECG monitoring during physical activities. Ongoing works concern online algorithms to detect and possibly compensate motion artifacts, by analyzing other signals acquired by the wearable device, like the bioelectrical impedance and MEMS accelerometer signals.

Acknowledgments

We would like to thank Marco Longoni for his invaluable support in the design and implementation of the demo and in the Bio2Bit dataset acquisition.

References

[1] İ. Güler, E. D. Übeylı, ECG beat classifier designed by combined neural network model, Pattern recognition 38 (2) (2005) 199–208.

[2] R. G. Afkhami, G. Azarnia, M. A. Tinati, Cardiac arrhythmia classification using statistical and mixture modeling features of ecg signals, Pattern Recognition Letters 70 (2016) 45–51.

[3] R. J. Martis, C. Chakraborty, A. K. Ray, A two-stage mechanism for registration and classification of ECG using gaussian mixture model, Pattern Recognition 42 (11) (2009) 2979–2988.

[4] S. Kara, M. Okandan, Atrial fibrillation classification with artificial neural networks, Pattern Recognition 40 (11) (2007) 2967–2973.

[5] S. Osowski, L. T. Hoai, T. Markiewicz, Support vector machine-based expert system for reliable heartbeat recognition, IEEE Transactions on Biomedical Engineering 51 (4) (2004) 582–589.

[6] Y. H. Hu, S. Palreddy, W. J. Tompkins, A patient-adaptable ECG beat classifier using a mixture of experts approach, IEEE Transactions on Biomedical Engineering 44 (9) (1997) 891–900.

[7] S. Kiranyaz, T. Ince, M. Gabbouj, Real-time patient-specific ECG classification by 1-d convolutional neural networks, IEEE Transactions on Biomedical Engineering 63 (3) (2016) 664–675.

[8] E. Keogh, J. Lin, A. Fu, Hot sax: Efficiently finding the most unusual time series subsequence, in: ICDM, 2005, pp. 226–233.

[9] MCOT Cardionet.
URL `http://www.cardionet.com`

[10] Smartheart.
URL `http://www.shl-telemedicine.com/portfolio/smartheart/`

[11] A. R. M. Forkan, I. Khalil, Z. Tari, S. Foufou, A. Bouras, A context-aware approach for long-term behavioural change detection and abnormality prediction in ambient assisted living, Pattern Recognition 48 (3) (2015) 628–641.

[12] W. Jiang, S. G. Kong, Block-based neural networks for personalized ecg signal classification, IEEE Transactions Neural Networks 18 (6) (2007) 1750–1761.

[13] P. de Chazal, M. O. Dwyer, R. B. Reilly, Automatic classification of heartbeats using ecg morphology and heartbeat interval features, IEEE Transactions on Biomedical Engineering 51 (7) (2004) 1196–1206.

[14] R. Hoekema, G. J. Uijen, A. Van Oosterom, Geometrical aspects of the interindividual variability of multilead ECG recordings, IEEE Transactions on Biomedical Engineering 48 (5) (2001) 551–559.

[15] D. Carrera, B. Rossi, D. Zambon, P. Fragneto, G. Boracchi, ECG monitoring in wearable devices by sparse models, in: ECML-PKDD, 2016, pp. 145–160.

[16] S. J. Pan, Q. Yang, A survey on transfer learning, IEEE Transactions on knowledge and data engineering 22 (10) (2010) 1345–1359.

[17] D. L. Mann, D. P. Zipes, P. Libby, R. O. Bonow, Braunwald's heart disease: a textbook of cardiovascular medicine, Elsevier Health Sciences, 2014.

[18] M. Aharon, M. Elad, A. Bruckstein, K-svd: An algorithm for designing overcomplete dictionaries for sparse representation, IEEE Transactions on Signal Processing 54 (11) (2006) 4311–4322.

[19] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online dictionary learning for sparse coding, in: ICML, 2009, pp. 689–696.

[20] A. L. Goldberger, L. A. 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, Circulation 101 (23) (2000) e215–e220.

[21] M. Longoni, D. Carrera, B. Rossi, P. Fragneto, M. Pessione, G. Boracchi, A wearable device for online and long-term ECG monitoring, in: IJCAI-ECAI, 2018.

[22] D. Carrera, B. Rossi, P. Fragneto, G. Boracchi, Domain adaptation for online ECG monitoring, in: ICDM, 2017, pp. 775–780.

[23] L. A. Saxon, Ubiquitous wireless ecg recording: a powerful tool physicians should embrace, Jour. Cardiovasc. Electr. 24 (4) (2013) 480–483.

[24] J. Wiens, J. V. Guttag, Active learning applied to patient-adaptive heartbeat classification, in: NIPS, 2010, pp. 2442–2450.

[25] M. Deng, C. Wang, M. Tang, T. Zheng, Extracting cardiac dynamics within ecg signal for human identification and cardiovascular diseases classification, Neural Networks 100 (2018) 70–83.

[26] H. Aidos, A. Lourenço, D. Batista, S. R. Bulò, A. Fred, Semi-supervised consensus clustering for ECG pathology classification, in: ECML-PKDD, 2015, pp. 150–164.

[27] R. D. Labati, E. Muñoz, V. Piuri, R. Sassi, F. Scotti, Deep-ecg: Convolutional neural networks for ecg biometric recognition, Pattern Recognition Letters.

[28] P. Malhotra, L. Vig, G. Shroff, P. Agarwal, Long short term memory networks for anomaly detection in time series, in: ESANN, 2015, pp. 89–94.

[29] Z. Sankari, H. Adeli, Heartsaver: a mobile cardiac monitoring system for auto-detection of atrial fibrillation, myocardial infarction, and atrioventricular block, Comput. Bio. Med. 41 (4) (2011) 211–220.

[30] M. Hadjem, O. Salem, F. Naït-Abdesselam, An ecg monitoring system for prediction of cardiac anomalies using wban, in: Healthcom, IEEE, 2014, pp. 441–446.

[31] M. Shao, C. Castillo, Z. Gu, Y. Fu, Low-rank transfer subspace learning, in: ICDM, 2012, pp. 1104–1109.

[32] L. A. Pereira, R. da Silva Torres, Semi-supervised transfer subspace for domain adaptation, Pattern Recognition 75 (2018) 235–249.

[33] A. M. Bruckstein, D. L. Donoho, M. Elad, From sparse solutions of systems of equations to sparse modeling of signals and images, SIAM review 51 (1) (2009) 34–81.

[34] X. Wei, Y. Li, H. Shen, W. Xiang, Y. L. Murphey, Joint learning sparsifying linear transformation for low-resolution image synthesis and recognition, Pattern Recognition 66 (2017) 412–424.

[35] G. Lin, M. Yang, J. Yang, L. Shen, W. Xie, Robust, discriminative and comprehensive dictionary learning for face recognition, Pattern Recognition 81 (2018) 341–356.

[36] Y. Chen, J. Su, Sparse embedded dictionary learning on face recognition, Pattern Recognition 64 (2017) 51–59.

[37] Y. Yuan, Y. Feng, X. Lu, Structured dictionary learning for abnormal event detection in crowded scenes, Pattern Recognition 73 (2018) 99–110.

[38] A. Adler, M. Elad, Y. Hel-Or, E. Rivlin, Sparse coding with anomaly detection, Journal of Signal Processing Systems 79 (2) (2015) 179–188.

[39] Q. Qiu, V. M. Patel, P. Turaga, R. Chellappa, Domain adaptive dictionary learning, in: ECCV, 2012, pp. 631–645.

[40] J. Ni, Q. Qiu, R. Chellappa, Subspace interpolation via dictionary learning for unsupervised domain adaptation, in: CVPR, 2013, pp. 692–699.

[41] S. Shekhar, V. M. Patel, H. V. Nguyen, R. Chellappa, Generalized domain-adaptive dictionaries, in: CVPR, 2013, pp. 361–368.

[42] B. Yang, A. J. Ma, P. C. Yuen, Learning domain-shared group-sparse representation for unsupervised domain adaptation, Pattern Recognition 81 (2018) 615–632.

[43] J. Pan, W. J. Tompkins, A real-time qrs detection algorithm, IEEE Transactions on Biomedical Engineering (3) (1985) 230–236.

[44] B. K. Natarajan, Sparse approximate solutions to linear systems, SIAM Journal on Computing 24 (2) (1995) 227–234.

[45] Y. C. Pati, R. Rezaiifar, P. S. Krishnaprasad, Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition, in: ACSSC, 1993, pp. 40–44.

[46] C. Lu, J. Shi, J. Jia, Online robust dictionary learning, in: CVPR, 2013, pp. 415–422.

[47] R. Tibshirani, Regression shrinkage and selection via the lasso, Journal of the Royal Statistical Society. Series B (Methodological) 58 (1996) 267–288.

[48] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Found. Trends® Mach. Learn. 3 (1) (2011) 1–122.

[49] Q. Liu, S. Wang, J. Luo, Y. Zhu, M. Ye, An augmented lagrangian approach to general dictionary learning for image denoising, Journal of Visual Communication and Image Representation 23 (5) (2012) 753–766.

[50] Y.-L. Yu, On decomposing the proximal map, in: NIPS, 2013, pp. 91–99.

[51] P. Gentile, M. Pessione, A. Suppa, A. Zampogna, F. Irrera, Embedded wearable integrating real-time processing of electromyography signals, in: Eurosensor, 2017, pp. 1–4.

[52] R. Rubinstein, M. Zibulevsky, M. Elad, Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit.

[53] STMicroelectronics, `http://www.st.com/en/evaluation-tools/nucleo-l476rg.html` (2016).

[54] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing 26 (1) (1978) 43–49.

[55] J. Bergstra, Y. Bengio, Random search for hyper-parameter optimization, Journal of Machine Learning Research 13 (Feb) (2012) 281–305.