Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Inter-beat Interval Estimation from Extremely Noisy Single Lead Electrocardiograms ⋆

Antoni Burguera *

* *Universitat de les Illes Balears, Ctra. Valldemossa Km. 7.5 Palma (Illes Balears), 07122 Spain e-mail: antoni.burguera@uib.es.*

Abstract: The advent of wearable recorders poses new challenges to electrocardiogram (ECG) analysis, such as robust feature extraction in front of long-term recordings with intervals of extreme noise. This paper proposes a robust approach to improve the estimates of one particular feature, the R-R interval (RRI), extracted by an arbitrary QRS detector operating in these scenarios. The proposal performs three steps. First, a voting schema is used to detect noisy intervals. Second, a rough estimate of the RRI evolution with time is obtained. Finally, this estimate is used to guide the Kalman filter in charge of refining the RRI estimates. Two groups of experiments have been performed. The first relies on 1674 real ECG corrupted with controlled amounts of noise. The second one tests our proposal using the *MIT-BIH Noise Stress Test Database*. Results show that influence of the initial error in our approach is small, leading to a large improvement in front of highly corrupted electrocardiograms at the cost of reducing the quality of the RRI estimates in absence of significant noise. Accordingly, the presented approach is suitable to process data obtained from portable ECG devices in which localized intervals of severe noise are present.

Keywords: Medical applications, ECG processing, Electrocardiography, R-R Interval, Signal processing

1. INTRODUCTION

One of the most widely used noninvasive medical procedures to measure the heart activity is the *electrocardiography*. This procedure records onto an *electrocardiogram* (ECG) the electrical changes on the skin due to the heart electrophysiological patterns.

The popularity of portable ECG recorders (Patel et al., 2012) is growing because its ease of use and compact format. Moreover, the advances in small low-cost microprocessor and microcomputer boards make it possible to perform on-line ECG processing. Consequently, wearable ECG technology is one of the main research lines nowadays (Elgendi et al., 2014; Baig et al., 2013).

The portable devices tend to be single lead thus being significantly less robust in front of noise. Moreover, since the goal of these devices is long-term monitoring, they have to deal with noise sources not present in controlled ECG scenarios. For example, extremely noisy intervals may appear while the patient is doing physical exercise or in presence of external electromagnetic interferences. In applications like epileptic seizure prediction or detection (Hashimoto et al., 2013), the seizures themselves may be responsible for extremely noisy intervals, jeopardizing the whole application purpose.

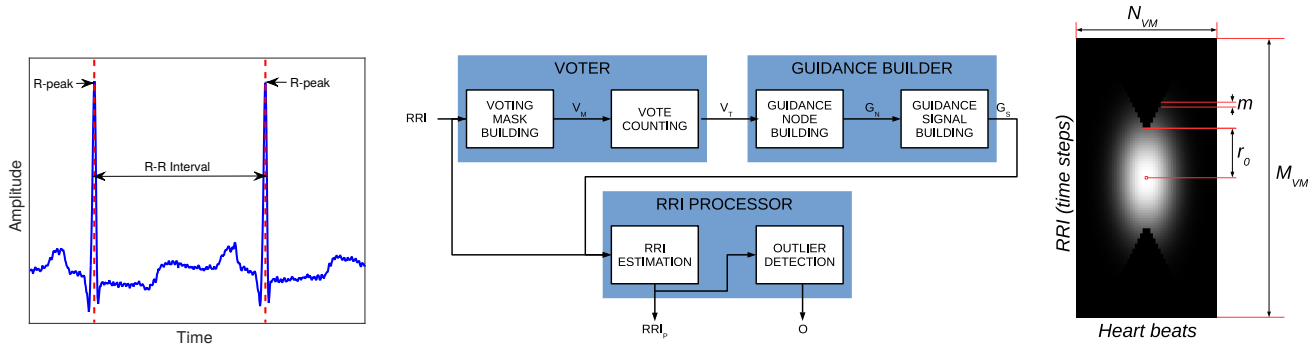
⋆ This work is partially supported by Ministry of Economy and Competitiveness under contracts DPI2017-86372-C3-3-R (AEI,FEDER,UE) and TIN2014-58662-R (AEI,FEDER,UE).

Current state of the art computer-based ECG analyzers exhibit exceptionally good results (Elgendi et al., 2017; Burguera, 2019) even in presence of moderate and high noise, being comparable to ECG labelled by experienced cardiologists. However, these approaches tend to fail, precisely, in the mentioned intervals of extreme noise.

This paper focuses on these cases. Our goal is to robustly estimate the inter-beat or *R-R Intervals* (RRI), defined as the time between consecutive heart beats (see Figure 1-a), in long-term, single lead ECG involving extremely noisy intervals. Thus, our approach constitutes a first step towards robust continuous ECG monitoring based on low-cost electrocardiographs.

More specifically, the proposal presented in this paper is composed of three main blocks that are executed sequentially. The first block is the *voter* and is in charge of isolating the intervals of extreme noise. The second block, the *guidance builder*, provides a rough estimate of the inter-beat intervals paying special attention to the detected intervals of extreme noise. Finally, this rough estimate is used by the *RRI processor* block to guide a *Kalman filter* (KF) through the noisy intervals. Figure 1-b summarizes the structure and introduces some basic notation and nomenclature that will be used throughout the document.

It is important to emphasize that our proposal is not aimed at detecting QRS complexes (Elgendi et al., 2017; Köhler et al., 2002) within an ECG. Instead, this study



(a) (b) (c)

Fig. 1. (a) The R-R interval. (b) Algorithm overview. (c) The voting mask V_M obtained for $M_{VM} = 0.3F_s$, $N_{VM} = 50$, $r_0 = 0.05F_s$ and $m = 0.005F_s$, being $F_s = 360Hz$. Brighter pixels correspond to higher values.

considers the inter-beat intervals computed by an arbitrary QRS detector as a signal that evolves with time and improves it globally. Thanks to that, a clean representation of the overall evolution of the instantaneous heart rate is obtained. Thus, our proposal can be used, among other, to improve the average heart rate estimates, usually obtained over a sliding window. As a side effect, our study also classifies each individual inter-beat interval as being normal, too large or too short. Accordingly, it provides crucial information that could be used either to improve QRS detection or to help cardiologists in detecting abnormal regions within an ECG.

2. THE VOTER

The voter is aimed at determining which parts of the RRI vector are likely to be correct and which ones are likely to be severly corrupted by noise. To this end, the RRI at each beat b, denoted by $RRI(b)$, votes those RRI that are acceptable from its own point of view by means of a voting mask (V_M). Afterwards, a vote counting procedure is performed resulting in a voting table (V_T). The most voted regions within this voting table are likely to correspond to the correct RRI.

The voting mask is a $M_{VM} \times N_{VM}$ matrix where each row represents a possible RRI value and each column represents a heart beat. Each cell of V_M holds a value of how likely the RRI it represents is correct if we assume that the RRI represented by the matrix central position is correct. In other words, each RRI assumes it is correct and evaluates positively those that are consistent with it. Let the RRI which is assumed to be correct be referred to as the *central RRI*.

To build the matrix, the following two criteria are used. The first criterion states that the allowed difference in value between an RRI and the central RRI increases with the difference in beat number. For example, given an RRI of 1 second at beat b, an RRI of 0.5 seconds may not be allowed at beat $b + 1$ but possible at beat $b + 2$. This criterion involves two parameters: r_0, which is the difference in RRI that is acceptable for the same beat that the central RRI, and m, which denotes how many units can increase the RRI for each beat of distance from the central position.

The second criterion states that those cells passing the first criterion will be assigned a value that follows a bivariate Normal distribution whose mean is at the central position. The covariance is selected so that the 4σ bound fully lies within the voting mask. Figure 1-c illustrates the voting mask.

Vote counting begins by building a voting table V_T. This table is a matrix where, similarly to V_M, each row represents a possible RRI value and each column represents a heart beat. However, contrarily to V_M, the voting table is global in the sense that it must cover all the data in the RRI vector. Thus, the number of columns is the number of beats N_B and the number of rows is the maximum RRI. All the V_T cells are initialized to zero.

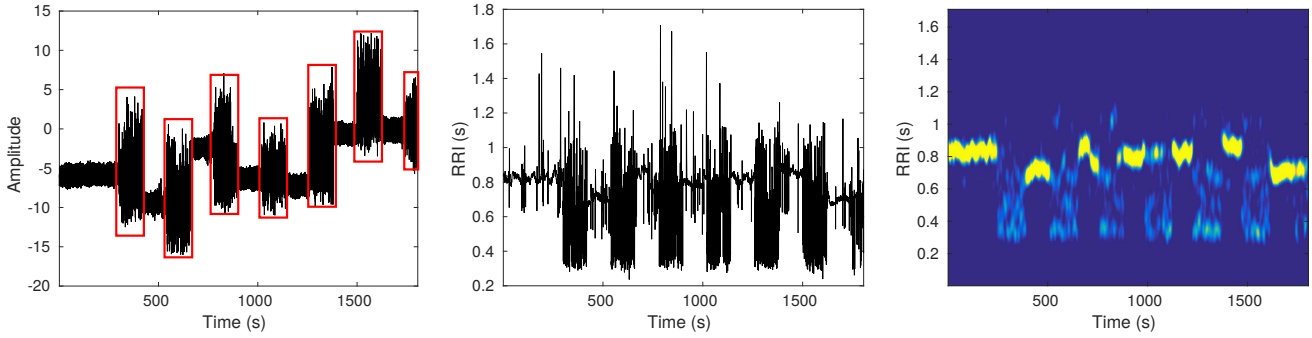
Afterwards, each $RRI(b)$ generates a vote matrix, which is summed to the corresponding cells of V_T. The emitted votes are computed by means of the voting mask assuming that each $RRI(b)$ is the central RRI of its V_M. In other words, each $RRI(b)$ votes those RRI that could be correct from its own point of view.

As an example, Figure 2-a shows an ECG with several intervals of severe noise marked with red rectangles. Figure 2-b shows the resulting RRI vector after processing the ECG with a QRS detector. As it can be observed, the RRI estimates become very noisy at the noisy ECG areas. Finally, Figure 2-c, which depicts the obtained voting table V_T, shows that the noise free areas are those where the votes accumulate.

3. THE GUIDANCE BUILDER

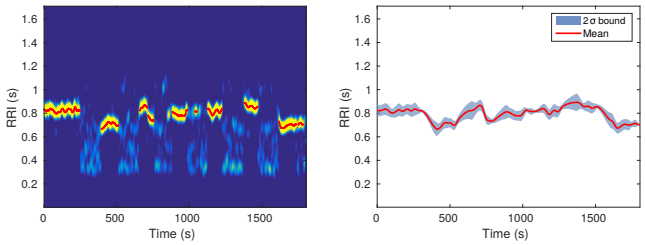
The goal of the guidance builder is to generate a signal that captures the main behavior of the true RRI of a given ECG by means of the aforedescribed voting table.

In order to decide whether a cell of the voting table has enough votes or not, a dynamic threshold τ is used (Otsu, 1979). In this way, $V_T(v, b) \geq \tau$ means that an RRI of duration v at heart beat b is consistent with the original RRI vector whilst $V_T(v, b) < \tau$ means that the original RRI vector does not provide enough information regarding this particular RRI. Let us define the set G_N of *guidance nodes* as the set of RRI that are the most consistent with the original RRI as follows:



(a) (b) (c)

Fig. 2. An example of the voter block. (a) ECG corresponding to record 118e00 from the nstdb database (Moody et al., 1984; Goldberger et al., 2000) with noise intervals marked with red rectangles. (b) The RRI obtained from a QRS detector. (c) The vote table. Brighter values correspond to the most voted RRI.



(a) (b)

Fig. 3. An example of the guidance builder. (a) The guidance nodes, depicted in red, over the corresponding vote table. (b) The guidance signal depicted by the mean (s_b) and the 2σ bound ($[s_b - 2\sigma_b, s_b + 2\sigma_b]$).

$$G_N = \{[v,b]^T \forall b \in \mathbb{N}, 0 \leq b < N_B, v = \underset{r}{\operatorname{argmax}}\, V_T(r,b) | v \geq \tau\} \quad (1)$$

As a result of this process, there are no guidance nodes at the beats with severe noise because, in these cases, the threshold τ will not be surpassed. Figure 3-a illustrates this idea.

The *guidance signal building* pursues two main goals. First, to fill the gaps corresponding to the beats not present in G_N. Second, to compute an error estimate for each beat of the resulting signal.

Filling the existing gaps in G_N is achieved by means of a standard interpolation technique. Our proposal is to use a shape-preserving piecewise cubic interpolation that guarantees continuity C^1. That is, it guarantees a differentiable function whose derivative is continuous.

As for the error estimates, the vote table is used similarly to the guidance node building step. In particular, for each column b of V_T, which represents a heart beat, we search the lowest (v_l) and the highest (v_h) rows, which represent an RRI, so that $V_T(v_l, b) \geq \tau$ and $V_T(v_h, b) \geq \tau$. Given the way in which V_T was built, it is reasonable to assume that given a guidance node $[v,b]^T$ the actual RRI value at this beat is likely to lie in the interval $[v_l, v_h]$. Our proposal is to use $\max(v_h - v, v - v_l)$ to approximate a 2σ bound for this guidance node. In other words, we assume that given a guidance node $[v,b]$, the actual RRI will be within the interval $v \pm \max(v_h - v, v - v_l)$ in the 95% of the cases.

However, similarly to G_N, gaps will appear in those regions of extreme noise where no RRI had enough votes. The uncertainty during these intervals grows and so the error estimate. Our proposal is to make it grow at a constant rate $\delta = 0.1$. This specific value has been obtained experimentally.

Let us define the *guidance signal* G_S as a vector containing both the interpolated values from the guidance nodes and the error estimates. Under the aforementioned assumptions, the value, either interpolated or existing in G_N, can be seen as the mean value of the signal and the error as two times the standard deviation.

More formally, $G_S(b) = [s_b, \sigma_b]^T, b \in \mathbb{N}, 0 \leq b < N_B$. The terms s_b and σ_b, which denote the mean value and the corresponding standard deviation, are computed as follows:

$$s_b = \begin{cases} v & \text{if } [v,b]^T \in G_N \\ interpolate(b, G_N) & \text{otherwise} \end{cases} \quad (2)$$

$$\sigma_b = \begin{cases} \frac{\max((\max(V_b) - s_b), (s_b - \min(V_b)))}{2} & \text{if } V_b \neq \emptyset \\ \sigma_{b-1} + \delta & \text{if } V_b = \emptyset \text{ and } b > 0 \\ K & \text{otherwise} \end{cases} \quad (3)$$

where K is a value used only if the first beat has no corresponding node in G_N (i.e. if the ECG was extremely noisy at the very beginning). As for $V_b = \{v | V_T(v,b) \geq \tau\}$, it is the set of RRI values at beat b that have enough votes.

It is important to emphasize that one guidance signal value $G_S(b)$ exists for each initial RRI value $RRI(b)$. Figure 3-b shows an example of guidance signal.

4. THE RRI PROCESSOR

The goal of the RRI processor is to improve the initial RRI vector, especially during the intervals of severe noise. As a side effect, the RRI processor also identifies RRI that are not statistically compatible with their neighbourhood and classifies each of them as a long beat or a short beat.

A common approach to estimate time-varying signals given a set of noisy measurements is the *Kalman Filter* (KF). However, it is well known that KF may not be able to recover from errors due to inconsistent measurements.

To alleviate this problem, our proposal is to use a KF to estimate the RRI and use the guidance signal G_S to prevent drift and allow recovery even in presence of inconsistent data. Next, the KF particularized to our specific application is described.

For each beat b, a KF operates in two steps: the *prediction step* and the *measurement update step*. The former provides a prior estimate by means of the a model of the system and the posterior estimate obtained for beat $b-1$. The latter incorporates the measurements to obtain the posterior estimate.

Our goal being to improve the initial RRI vector, let the KF state vector at beat b represent an RRI estimate at that beat. Also, as a KF works with Normal distributions expressed by its mean and covariance, let $\overline{x}_b = N(\overline{\mu}_b, \overline{\Sigma}_b)$ denote the prior estimate at beat b and $x_b = N(\mu_b, \Sigma_b)$ denote the posterior, being μ_b and Σ_b the mean and the covariance respectively.

The filter is initialized by assuming that the first RRI in the RRI vector is perfectly known. That is, $x_0 = RRI(0)$ and $\Sigma_0 = 0$. After the initialization, the prediction and the measurement update steps are executed until the whole RRI vector has been processed.

During the prediction step, we assume that the mean RRI estimate obtained at beat $b-1$ will not change at beat b, but the covariance will increase to take into account possible changes. That is, the prediction or process equations are $\overline{\mu}_b = \mu_{b-1}$ and $\overline{\Sigma}_b = \Sigma_{b-1} + Q$, where Q is the covariance matrix of the process noise and represents how much can the current RRI change with respect to the previous one.

During the measurement update, both $RRI(b)$ and $G_S(b)$ will be used to obtain the posterior. In particular, the measurement update equations are as follows:

$$K_b = \overline{\Sigma}_b H' (R_b + H\overline{\Sigma}_b H')^{-1} \quad (4)$$

$$\mu_b = \hat{\mu}_b + K_b(z_b - H\overline{\mu}_b) \quad (5)$$

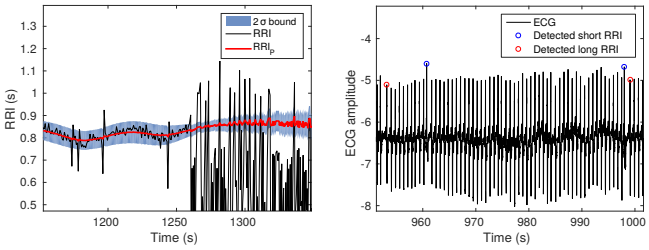
$$\Sigma_b = (1 - K_b H)\overline{\Sigma}_b \quad (6)$$

where z_b is the measurement vector, H is the observation matrix and R_b is the measurement covariance.

Since our proposal is to use both the initial RRI and the guidance as measurements, $z_b = [RRI(b), s_b]^T$ where s_b comes from $G_S(b) = [s_b, \sigma_b]^T$. Accordingly, the observation matrix H is $[1, 1]^T$. As for the measurement covariance, our proposal is $R_b = diag(\frac{0.2F_s}{2}, \sigma_b^2)$. That is, we use the same error estimate used during the prediction for RRI and the computed σ_b for the guidance signal.

By iterating through the prediction and the measurement update steps, the processed RRI or RRI_P is obtained so that $RRI_P(b) = x_b$.

Figure 4-a exemplifies the obtained results. As it can be observed, RRI_P smoothly follows the original RRI except for the intervals of extreme noise, where other values are proposed. Also, the obtained Σ_b seems to provide an appropriate 2σ bound, as most of the RRI corresponding to noise free regions lie within these boundaries.



(a) (b)

Fig. 4. (a) RRI estimation. (b) Outlier detection.

Under the Gaussian assumption performed by the KF, $RRI(b)$ is expected to be within the interval $[\overline{\mu}_b - 2\sqrt{\overline{\Sigma}_b}, \overline{\mu}_b + 2\sqrt{\overline{\Sigma}_b}]$ in the 95% of the cases. Thus, an RRI below $\overline{\mu}_b - 2\sqrt{\overline{\Sigma}_b}$ is likely to be too short and an RRI larger than $\overline{\mu}_b + 2\sqrt{\overline{\Sigma}_b}$ is probably too large. Our proposal makes it possible to detect some beats requiring further inspection. This outlier detection could be used to help cardiologists to detect heart problems or to improve the detection rate of an existing QRS detector.

Figure 4-b shows an example of detected outliers. In this case, the system properly detected beats that are too short or too large with respect to their neighborhood. Thus, using the outlier detection to help cardiologists in detecting abnormalities also seems reasonable.

5. RESULTS AND DISCUSSION

Two groups of experiments have been performed in order to test the validity of our proposal. The first group is a stress test. In this case, a wide range of real ECG databases has been used and different amounts of controlled noise have been added in order to measure the quality of our proposal as a function of the existing noise. The second group evaluates our approach by means of a standard ECG noise stress test database. In all cases, the initial RRI is computed from the R-peaks provided by the QRS detector described by Burguera (2019), whose source code is publicly available at `https://github.com/aburguera/QRS_DETECT`.

5.1 Stress test

To evaluate the quality of our proposal in front of known amounts of noise, 8 different ECG databases (Goldberger et al., 2000) have been used. Each database is composed of different records and channels, leading to a total of 1674 different ECG involving a wide range of heart behaviours, sampling frequencies, leads, ECG lengths and initial noise conditions. Moreover, since these databases have been properly tagged by expert cardiologists, the ground truth is available.

The amount of added noise is controlled by two parameters: the number of intervals (NI) and the interval length (IL). The number of intervals defines with how many disjoint intervals of extreme noise is the ECG corrupted. The interval length states the duration of the noisy intervals as a percentage of the ECG length. Three different NI (one, three and five) and two different IL (5% and 10%) have been tested. Accordingly, the best possible configuration

Table 1. Mean error of the initial (e_I) and the processed (e_P) RRI, standard deviation of the initial (σ_I) and the processed (σ_P) RRI and percentage of processed error with respect to the initial one (%e). Gray cells denote situations in which e_P is below e_I.

	NI	1		3		5	
DB	IL	5%	10%	5%	10%	5%	10%
AFTDB	e_I/e_P	17.1/18.5	18.5/19.2	21.9/19.8	26.5/22.1	26.8/21.5	34.6/25.8
	σ_I/σ_P	17.1/13.6	16.6/13.3	16.1/13.2	14.7/12.5	15.4/12.8	13.4/12.0
	%e	108.7%	103.3%	90.6%	83.1%	80.2%	74.6%
Fantasia	e_I/e_P	3.1/3.7	5.7/6.0	8.4/6.4	16.3/13.2	13.7/9.2	27.0/20.5
	σ_I/σ_P	1.0/2.5	1.1/3.3	1.3/4.3	2.1/7.1	1.8/6.5	3.4/11.1
	%e	120.4%	105.1%	76.7%	81.0%	67.0%	76.0%
INCARTDB	e_I/e_P	7.4/10.3	9.4/11.2	11.6/11.4	17.9/14.1	15.9/12.5	26.3/17.3
	σ_I/σ_P	8.7/10.1	8.3/10.1	7.9/10.3	6.9/10.8	7.2/10.6	5.8/12.1
	%e	140.1%	118.5%	98.2%	78.9%	79.0%	65.9%
QTDB	e_I/e_P	11.3/11.8	13.6/12.4	16.0/12.4	23.1/14.4	20.8/13.2	32.4/16.6
	σ_I/σ_P	33.4/33.7	33.5/33.8	33.6/33.8	34.1/34.3	33.9/34.0	34.6/35.4
	%e	104.5%	91.0%	77.6%	62.2%	63.4%	51.3%
SVDB	e_I/e_P	9.7/11.8	11.4/12.8	13.2/13.0	18.5/16.1	16.8/14.2	25.6/19.6
	σ_I/σ_P	10.6/11.5	10.4/11.3	10.1/11.4	9.4/11.1	9.6/11.5	8.6/11.2
	%e	122.4%	112.4%	98.0%	87.2%	84.4%	76.4%
EDB	e_I/e_P	21.2/21.0	23.4/22.5	25.5/22.4	32.1/26.8	29.9/23.8	40.9/31.2
	σ_I/σ_P	47.2/47.0	47.3/47.3	47.5/47.4	48.1/48.3	47.9/47.7	48.8/49.5
	%e	99.1%	96.1%	87.7%	83.3%	79.4%	76.2%
MITDB	e_I/e_P	10.5/12.3	12.5/13.1	14.6/13.2	20.7/15.4	18.7/14.2	28.8/17.9
	σ_I/σ_P	8.5/9.8	8.2/10.0	7.8/10.2	7.0/11.2	7.3/10.7	6.2/13.0
	%e	117.0%	104.2%	90.4%	74.8%	75.9%	62.2%
STDB	e_I/e_P	4.4/2.9	6.5/3.3	9.0/3.4	16.0/4.9	13.7/3.8	24.9/6.1
	σ_I/σ_P	4.7/3.0	4.6/3.3	4.6/3.8	4.5/4.9	4.2/4.1	4.1/5.8
	%e	65.4%	50.1%	37.7%	30.4%	27.5%	24.6%
Global	e_I/e_P	11.2/12.7	13.2/13.6	15.6/13.8	21.6/16.5	20.0/15.0	30.0/19.8
	σ_I/σ_P	21.1/20.8	20.9/20.9	20.9/20.9	20.7/21.3	20.9/21.1	20.6/22.2
	%e	112.9%	102.7%	88.4%	76.6%	75.1%	66.2%

is 1 interval of a 5% of the ECG length and the worst scenario corresponds to five intervals of a 10% of duration, meaning that half of the ECG is extremely corrupted.

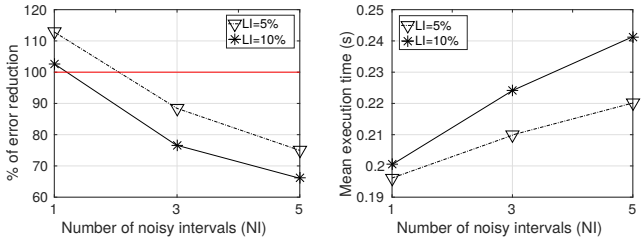
The synthetic noise used for each corrupted interval is additive with two components. First, a simulated baseline drift involving changes in amplitude up to the maximum amplitude difference within the clean ECG interval. Second, a high frequency noise corrupting each sample with random Gaussian values extracted with zero mean and a standard deviation which is also the maximum amplitude difference within the clean ECG interval. The corrupted ECG has been processed by a QRS detector and the initial RRI has been computed from its output. Then, our proposal has been executed and RRI_P has been evaluated.

This process has been repeated 100 times per combination of parameters in order to obtain statistically significant results. For each of the 100 trials, the disjoint intervals of noise have been randomly placed over the ECG according to a uniform distribution. As for the error of the processed RRI, it is computed as follows:

$$e_P = \frac{\sum_{b=0}^{N_B-1} |RRI_P(b) - RRI_{GT}(b)|}{N_B} \quad (7)$$

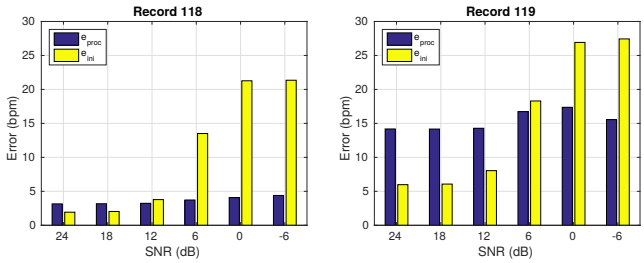
where RRI_{GT} denotes the *ground truth RRI*, which is the correct RRI. In other words, the e_P measures the mean error in amplitude of RRI_P per heart beat. The error e_I, which measures the mean error in amplitude of the initial RRI has also been computed analogously.

Table 1 summarizes the mean and the standard deviation of the errors corresponding to the initial and the processed RRI as a function of the number of intervals and the interval length. The improvement due to our proposal increases with the amount of noise. That is, the worse is the ECG the larger is the improvement of RRI_P with respect to RRI_I. Also, in most cases the standard deviation



(a) (b)

Fig. 5. (a) Global percentages of error after processing the RRI with respect to the error before processing it depending on the number and length of the extreme noise intervals. Percentages below 100% corresponds to improvements. (b) Mean execution times depending on the number and length of the extreme noise intervals.



(a) (b)

Fig. 6. Summary of the errors before (e_{ini}) and after (e_{proc}) the RRI processing for the MIT-BIH Noise Stress Test Database (a) record 118 and (b) record 119.

of the error after executing our proposal is similar to the standard deviation before executing it. This suggests that our approach is barely influenced by the specific distribution of the noise over the ECG. Unfortunately, for low amounts of noise the resulting RRI_P is slightly worse than the original one. However, results also show that our proposal is able to reduce the initial error to less than one fourth in the best case and to an overall 66.2% in the worst scenario. These results, are summarized in Figure 5-a, where it can be observed that the quality of RRI_P is better than the original RRI from three intervals onward.

In order to evaluate the efficiency of the presented approach, the execution time was measured on an i7 CPU at 3.1GHz using a single CPU kernel and a Matlab implementation. The results, summarized in Figure 5-b, show that even though the execution time increases with the amount of noise, the absolute difference between the best and the worst case barely surpasses 45 ms, being the mean ECG duration of 45 minutes. In particular, our proposal spent an average of 4.3 ms to process one minute of ECG in the best case and 5.4 ms in the worst case.

5.2 Standard database

The *MIT-BIH Noise Stress Test Database* (nstdb) (Moody et al., 1984; Goldberger et al., 2000) has also been used to evaluate the presented approach. This database was constructed from two clean half hour ECGs, named record 118 and record 119 composed of two channels each and made using standard recorders. Then, six different amounts of actual noise, typical of ambulatory ECGs, was added to

Table 2. Errors before (e_I) and after (e_P) the RRI processing for each record in the MIT-BIH Noise Stress Test Database. The shown values consider the two existing channels of each record and noise combination. Gray cells denote situations in which e_P is below e_I

Record	SNR	e_I	e_P	$100 \cdot \frac{e_P}{e_I}$
118	**24 dB**	1.930	3.156	163.478%
	18 dB	2.038	3.180	156.028%
	12 dB	3.784	3.238	85.567%
	6 dB	13.506	3.726	27.590%
	0 dB	21.271	4.081	19.187%
	-6 dB	21.352	4.393	20.574%
	GLOBAL	**11.012**	**3.649**	**33.135%**
119	**24 dB**	5.979	14.182	237.201%
	18 dB	6.068	14.166	233.444%
	12 dB	8.044	14.284	177.577%
	6 dB	18.298	16.727	91.417%
	0 dB	26.919	17.370	64.527%
	-6 dB	27.431	15.556	56.708%
	GLOBAL	**16.089**	**15.456**	**96.065%**
GLOBAL		**13.407**	**9.220**	**68.770%**

each channel of each clean record, leading to 24 different datasets. The noise added contained baseline wander as well as muscular and electrode motion artifacts. The specific signal to noise ratios (SNR) during the noisy segments of the records are 24 dB (low noise), 18 dB, 12 dB, 6 dB, 0 dB and -6 dB (high noise). This noise was not added by us, but it is part of the database, whose purpose is, precisely, to test ECG processors in front of noisy intervals. As an example, the ECG shown in Figure 2-a belongs to this database.

Table 2, which summarizes the results, shows an error reduction trend similar to the one obtained during the stress test: The improvement or RRI_P with respect to RRI_I increases with the amount of noise, our proposal leading to estimates worse than the initial ones only in presence of reduced amounts of noise. Figures 6-a and 6-b summarize these results in terms of *beats per minute* (bpm). Even with record 119, in which our approach exhibits less performance than with record 118, the final error remains almost constant independently of the amount of noise.

As for time consumption, our approach took a mean of 0.25 seconds per dataset with a standard deviation of 0.03 seconds, which implies a mean processing time of 0.11 ms per heart beat.

6. CONCLUSION

A method to improve the RRI estimates obtained from an ECG involving intervals of extreme noise has been presented. The proposal has been evaluated by two groups of experiments involving a wide range of ECG and noise conditions. The results show that the presented approach significantly improves the RRI estimates in presence of large amounts of noise. This being the kind of data that can be expected from portable ECG recorders, our approach is suitable to improve the RRI estimates using wearable technology.

Time consumption has shown to be very small. In particular, a Matlab implementation running on a standard laptop computer processed in average one minute of ECG in less than 6 ms. Portable devices with reduced computational capabilities could benefit from this execution speed.

As a side effect, our proposal is able to identify outliers, classifying them as short or long RRI. Accordingly, an interesting line of further research is to use these outliers both to improve the underlying QRS detection and to help cardiologists to find abnormal regions within an ECG.

REFERENCES

Baig, M.M., Gholamhosseini, H., and Connolly, M.J. (2013). A comprehensive survey of wearable and wireless ECG monitoring systems for older adults. *Medical and Biological Engineering and Computing*, 51(5), 485–495. doi:10.1007/s11517-012-1021-6.

Burguera, A. (2019). Fast QRS detection and ECG compression based on signal structural analysis. *IEEE Journal of Biomedical and Health Informatics*, 23(1), 123–131.

Elgendi, M., Eskofier, B., Dokos, S., and Abbott, D. (2014). Revisiting QRS detection methodologies for portable, wearable, battery-operated, and wireless ECG systems. *PLoS ONE*, 9(1). doi:10.1371/journal.pone.0084018.

Elgendi, M., Mohamed, A., and Ward, R. (2017). Efficient ECG Compression and QRS Detection for E-Health Applications. *Nature (Scientific Reports)*, 7(1), 459. doi:10.1038/s41598-017-00540-x.

Goldberger, A.L., Amaral, L.A.N., Glass, L., Hausdorff, J.M., Ivanov, P.C., Mark, R.G., Mietus, J.E., Moody, G.B., Peng, C.K., and Stanley, H.E. (2000). PhysioBank, PhysioToolkit, and PhysioNet Components of a New Research Resource for Complex Physiologic Signals. *Circulation*, 101(23), e215–e220.

Hashimoto, H., Fujiwara, K., Suzuki, Y., Miyajima, M., Yamakawa, T., Kano, M., Maehara, T., Ohta, K., Sasano, T., Matsuura, M., and Matsushima, E. (2013). Heart rate variability features for epilepsy seizure prediction. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2013*. doi:10.1109/APSIPA.2013.6694240.

Köhler, B.U., Hennig, C., and Orglmeister, R. (2002). The Principles of Software QRS Detection. *IEEE Engineering in Medicine and Biology Magazine*, 21(1), 42–57.

Moody, G., Muldrow, W., and Mark, R. (1984). A noise stress test for arrhythmia detectors. *Computers in Cardiology*, (11), 381–384.

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66. doi:10.1109/TSMC.1979.4310076. URL `http://ieeexplore.ieee.org/document/4310076/`.

Patel, A.M., Gakare, P.K., and Cheeran, A.N. (2012). Real Time ECG Feature Extraction and Arrhythmia Detection on a Mobile Platform. *International Journal of Computer Applications*, 44(23), 40–45. doi:10.5120-6432-8840.