Novel Real-Time Low-Complexity QRS Complex Detector Based on Adaptive Thresholding

Raquel Gutiérrez-Rivas, *Student Member, IEEE*, J. Jesús García, *Member, IEEE*, William P. Marnane, *Member, IEEE*, and Álvaro Hernández, *Member, IEEE*

***Abstract*— Over the years, several QRS complex detection algorithms have been proposed with different features, but the remaining problem is their implementation in low-cost portable platforms for real-time applications, where hardware resources are limited, still providing the accuracy level required for medical applications. The proposed algorithm copes at the same time with both requirements: 1) accuracy and 2) low resource consumption. In this paper, a real-time QRS complex detector is proposed. This algorithm is based on a differentiation at the pre-processing stage combined with a dynamic threshold to detect R peaks. The thresholding stage is based on a finite-state machine, which modifies the threshold value according to the evolution of the signal and the previously detected peak. It has been evaluated on several databases, including the standard ones, thus resulting sensitivities and positive predictivities better than 99.3%. In order to analyze the computational complexity of the algorithm, it has been compared with the well-known Pan and Tompkins' algorithm. As a result, the proposed detector achieves a reduction in processing time of almost 50% by using only the 25% of hardware resources (memory, adders, and multipliers).**

***Index Terms*— ECG signal, real-time QRS complex detection, remote monitoring systems.**

I. Introduction

QRS COMPLEX is the most important waveform within electrocardiogram (ECG), serving as the basis for the diagnosis of different pathologies related to heart. For example, the continuous observation of the heart rate permits the detection of cardiac diseases such as arrhythmias [1], making possible the implementation of an alarm system, which can reduce considerably the mortality by linking the monitoring device with a positioning system as is proposed in [2]. In addition, the detection of the QRS complex leads to the calculation of the Heart Rate Variability (HRV), and due to its relationship with several physiological systems [3] (vasomotor, respiratory, central nervous, thermoregulatory, etc.), it can be used as a diagnostic tool or an information system about their performance. Thus, it is possible to detect other diseases in a non-invasive way through the real-time analysis of the HRV, as Bsoul et al. proposed in [4], where apnoea episodes are detected by analysing the respiration patterns. Another example of this analysis of the ECG is the evaluation of the HRV to detect epileptic seizures, described by Massé et al. [5].

All these applications have some similarities. Firstly, they require a reliable QRS complex detector. Secondly, the detector has to provide real-time information about the heart rate. And finally, as they are using portable devices for implementing the QRS complex detector together with all the algorithms related to the application, they should be as much efficient as possible from the point of view of computational complexity and power consumption.

Due to its inherent importance, software QRS detection has been a research topic for more than four decades. As a result of this interest, plenty of algorithms have been published that reflect the evolution of computer technology [6]. At the beginning the computational load determined the complexity and therefore the performance of algorithms, whereas recent researches focus on the performance as computers are becoming faster, more powerful and more reliable. Nevertheless, with the appearance of the wearable technology, where low-power battery-driven devices are required, this trend is changing to the original idea of developing low computational load algorithms [7], [8] as is stated in [9].

Thanks to the evolution of technologies, there are some new alternatives to perform a real-time analysis of the ECG signal. One of the latest is the use of cloud computing as is proposed by Xia et al. in [10]. However, this proposal needs a permanent internet connection to achieve truly real-time results. Another possibility is the use of a host system to analyse the ECG signal measured by a remote device, in order to avoid the restrictions derived from its limited availability of resources. In this case, a new problem arises related to the communication issues, as the ECG signal has to be sent continuously to the analysis unit. So, to avoid communication problems, it is necessary to compress, transmit and reconstruct all the data [11]. Another approach is the design of an *ad hoc* ECG monitoring and analysis device, as the one proposed by Chou et al. in [12]. In this work, the authors carried out

Manuscript received June 11, 2015; accepted June 25, 2015. Date of publication June 29, 2015; date of current version August 19, 2015. This work was supported in part by the Spanish Ministry of Economy and Competitiveness through the LORIS Project and DISSETSOC Project under Grant TIN2012-38080-C04-01 and Grant TEC2012-38058-C03-03 and in part by the University of Alcalá through the FPI Program. The associate editor coordinating the review of this paper and approving it for publication was Dr. Chirasree Roychaudhuri.

R. Gutiérrez-Rivas is with the Department of Electronics, University of Alcalá, Madrid 28801, Spain, and also with the Department of Electrical and Electronic Engineering, University College Cork, Cork, Ireland (e-mail: raquel.gutierrez@edu.uah.es).

J. J. García, and Á. Hernández are with the Department of Electronics, University of Alcalá, Madrid 28801, Spain (e-mail: jjesus.garcia@uah.es; alvaro.hernandez@uah.es).

W. P. Marnane is with the Department of Electrical and Electronic Engineering, University College Cork, Cork, Ireland (e-mail: l.marnane@ucc.ie).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSEN.2015.2450773

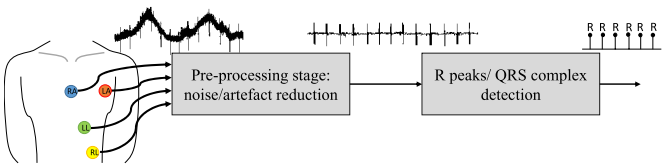

Fig. 1. Typical structure of a QRS complex detection algorithm.

a SoC solution based on UMC 90 nm CMOS technology. The system acquires the ECG, detects the R peaks and analyses the HRV signal, reducing the power consumption, the design cost and the final size.

In this work, we propose a novel real-time QRS complex detection algorithm based on a differentiation at the pre-processing stage combined with a dynamic threshold to detect the R peaks. Due to its low computational cost it is very suitable for being implemented in devices with very-reduced computational resources, where a high accuracy is required. The paper is organized as follows: Section 2 describes the most commonly used methods to perform QRS complex detection; Section 3 summarizes the features of the databases (standard and no-standard) used to train and test the proposed algorithm; Section 4 describes the novel proposed algorithm; some experimental results are shown in Section 5; and, finally, some conclusions are discussed in Section 6.

II. QRS Complex Detection Structure

The main challenge facing any QRS detector is to perform accurate heartbeat detection even with the presence of noises and artefacts. Trying to reach this objective, the detection algorithms are generally based on two main blocks (Fig. 1): a pre-processing stage, which attempts to reduce or remove most part of the noise; and the detection stage, in which the R peaks of the ECG signal are detected. Most of the R peak detection algorithms could be classified depending on the techniques that the authors propose for implementing each block. The most common pre-processing techniques are:

- Wavelet Transform (WT): Dyadic [13], Discrete (DWT) [14], Quadratic Spline (QSWT) [15], Continuous (CWT) [16]. This transform is often used together with a filtering stage in order to reduce high and/or low frequency noises.
- Differentiation, integration, squaring, etc. of the ECG signal: These techniques are based on the well-known Pan and Tompkins algorithm [17]. As they are the simplest techniques, they require a more complex detection technique to avoid the presence of the non-removed artefacts and the use of a less noisy acquisition platform. Some examples can be found in [18]–[20].
- Hilbert Transform: It provides a zero-crossing every time there is an inflexion in the raw signal. However, the features of several artefacts that are commonly present in the ECG signal could be mistaken as R peaks. For this reason, it is normally used together with another pre-processing stage which can be composed by a cascade of differentiations, integrations, etc., [21]; or they can be applied after [22] or before [23] a WT.
- Empirical Mode Decomposition (EMD) [24]: EMD decomposes the signal into a sum of oscillatory functions (Intrinsic Mode Functions, IMFs), then the IMFs, which do not contain information about the ECG signal are removed and the ECG is reconstructed with the non-removed IMFs. Some authors combine this technique with WT [25] in order to improve the detection performance, as well as with a Hilbert Transform stage [26].

With regard to the detection stage, the most used techniques are: analysis of the ECG morphology (slope analysis, zero-crossing detections, etc.); search for the maxima; and single, dual or even triple dynamic or adaptive thresholds to detect the amplitude of the ECG signal and the time between R peaks. According to the published works, taking into account both stages, the most common combination is the use of the wavelet decomposition with a single dynamic threshold or with dual dynamic threshold [27] to detect the R peaks. There are some online QRS detectors that only use one dynamic threshold [28], [29], obtaining an accurate detection. Their results can be improved if the Pan-Tompkins' search-back technique is included, as in [30].

III. Benchmark ECG Databases

Several standard ECG databases are available to evaluate QRS complex detection algorithms. The use of these well-annotated and validated databases provides reproducible and comparable results in terms of accuracy. These databases contain a large variety of selected ECG signals that allow the testing of the algorithms under different conditions, from records with clear R peaks and few artefacts to others with abnormal shapes, noise and lots of artefacts that make an accurate detection difficult. These databases are introduced now in this manuscript since they are necessary for tuning some parameters of the proposed algorithm.

In order to compare the performance of the proposed algorithm with previous works, it will be tested by using the data provided by MIT-BIH Arrhythmia Database (MITDB) [31]. This database contains 48 ECG recordings, each with a duration of 30 minutes, at a sampling rate F_S of 360Hz. This set represents different types of phenomena that arrhythmia can provoke in ECG signals. The MIT-BIH Normal Sinus Rhythm Database (NSRDB) will be used as well. This database is composed of 130-minutes long ECG signals from 18 healthy adults (aged from 20 to 50), at a sampling frequency of 128Hz. Furthermore, the authors have obtained another database (Allergy Database, ADB) from the Pediatrics Section of Cork University Hospital (Cork, Ireland). It is composed of 24 ECG signals with different lengths (from 13 minutes to more than 130 minutes), sampled at 256Hz. All the subjects in this database are children (from 7 months to 10 years old), and most of these ECG signals are affected by several motion artefacts. In addition, three more databases have been obtained through the Physiobank web site [32]: ApneaECG, Fantasia and Challenge 2014 databases.

There are several reasons to use so many databases: firstly, if the algorithm is successfully checked with as much data as possible, a higher level of robustness can be concluded; secondly, the acquisition conditions are different for each database, so it allows the analysis of the dependency between

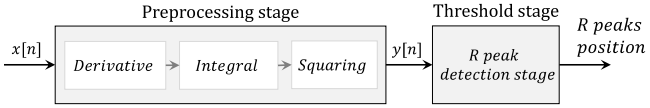

Fig. 2. Block diagram of the proposed QRS complex detection algorithm.

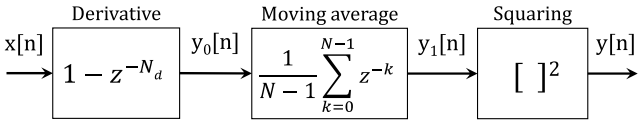

Fig. 3. Block diagram of the preprocessing stage.

these conditions and the algorithm parameters; and finally, most authors validate their algorithms with the MIT-BIH Arrhythmia Database, so it is needed to use this standard database for comparing the results with previous works.

IV. Real-Time QRS Complex Detection Algorithm

As explained before, the algorithm designed here should serve as a basis for applications that require real-time information about the R peak positions, as well as the implementation on platforms with reduced resource availability. For this reason, every stage in the algorithm has been designed trying to minimize the requirements used by our algorithm. It is important to keep in mind that this kind of algorithms is often oriented to mobile and portable applications, where the resource needs, as well as the associated power consumption, become a significant issue involved in the design of any future real-time implementation. In this way we manage to reduce the power consumed by the devices on which this algorithm will be implemented in the future. Fig. 2 shows the block diagram of the proposed R-peak detection algorithm. As has been previously mentioned, it is composed of two main blocks: the pre-processing stage and the thresholding one.

For a better understanding of the proposed algorithm, the results from every stage will be represented by using the ECG record for the subject 108m in the MIT-BIH Arrhythmia Database (see top graph in Fig. 6).

A. Pre-Processing Stage

Fig. 3 shows the block diagram of the pre-processing stage. In order to reduce low-frequency noise, the first step of the pre-processing stage is the derivation of the input ECG signal $x[n]$ according to (1). This process mainly reduces the wandering baseline effect. Then, an integration operation is carried out to remove the high-frequency artefacts from the signal $y_0[n]$, following (2). As it shows later, the value of N is very small, so this stage is acting as a low pass filter (Moving Average). The width N of the integration window varies depending on the sampling frequency F_S. Finally, in order to emphasize the R peaks, every sample $y_1[n]$ is squared (3). The signals obtained from these processes are shown in Fig 6.

$$y_0[n] = x[n] - x[n - N_d] \tag{1}$$

$$y_1[n] = \frac{1}{N-1}\sum_{k=0}^{N-1} y_0[n-k] \tag{2}$$

$$y[n] = (y_1[n])^2 \tag{3}$$

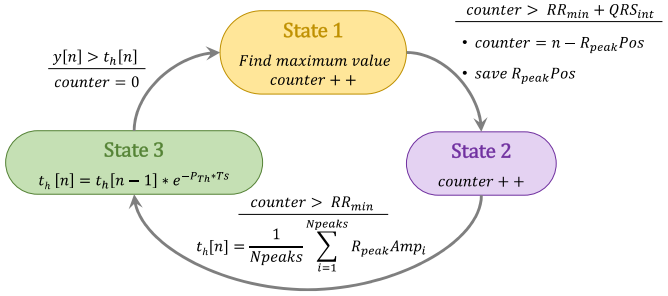

Fig. 4. Block diagram of the proposed FSM.

B. Threshold Stage

For this stage, it is very common to use an adaptive threshold, as most of the recently published real-time QRS detection algorithms. This adaptation capability of the threshold is essential when the pre-processing stage cannot remove all the artefacts and mainly in those signals with large T waves (to know more about the different ECG waves, peaks and intervals, see [33]), since they could be misclassified as R peaks. In this proposal, the threshold value is controlled by a finite state machine (FSM, Fig. 4), according to the following 3 states:

- *State 1 (Looking for a Maximum Peak):* During a time interval equal to the minimum feasible RR interval RR_{min} plus the standard duration of a QRS complex QRS_{int} (typically 60ms), the algorithm searches for the maximum peak of the signal. This maximum will be classified as an R peak. The machine changes to the following state when the interval $RR_{min}+QRS_{int}$ ends. In this case, the value RR_{min} is the RR interval corresponding to a heart rate of 300bpm (beats per minute), so $RR_{min} = 200$ms (it is considered that nobody can have a heart rate higher than 300bpm). At the end of this state the threshold amplitude is the mean $R_{peak}Amp$ of the amplitudes of all the R peaks found.
- *State 2 (Waiting State):* The duration of this state depends on the position $R_{peak}Pos$ where the R peak was found in State 1. The FSM is waiting for a time equal to RR_{min} less the time between the position of the last R peak and the end of State 1. Through it, false detections can be avoided during the interval RR_{min} after the last peak was detected, as this is the period when a long T wave could be misclassified as R peak.
- *State 3 (Threshold Decreasing):* When State 2 finishes, the initial value of the threshold $t_h[n]$ is computed as the mean value $R_{peak}Amp$ of all the previous detected R peaks. In this state, the threshold value $t_h[n]$ is reduced with every new sample from the input ECG signal $x[n]$, according to (4). The value of the parameter P_{TH} varies depending on the sampling frequency F_S as is explained below. This state ends when the level of the ECG signal $x[n]$ is higher than the threshold value $t_h[n]$.

$$t_h[n] = t_h[n-1] \cdot e^{-\frac{P_{TH}}{f_S}} \tag{4}$$

Figure 5. depicts the correspondence between these states and the phase of the ECG signal.

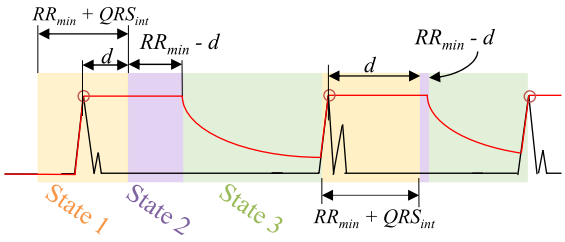

Fig. 5. Correspondence between the FSM states and the ECG phases.

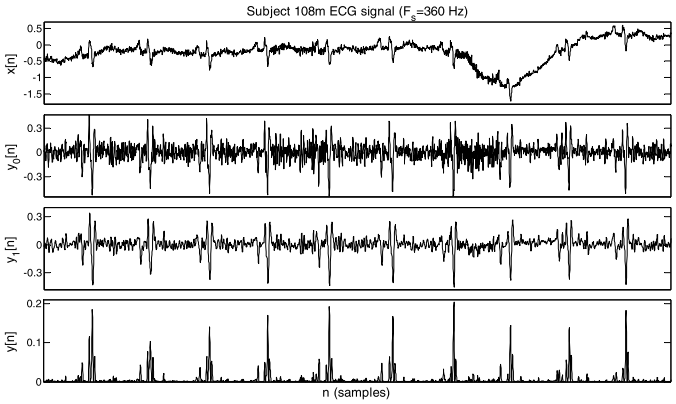

Fig. 6. Preprocessing results for signals $x[n]$, $y_0[n]$, $y_1[n]$ and $y[n]$.

C. Parameter Selection

The proposed QRS complex detection algorithm is based on three parameters: the derivative parameter N_d; the length N of the integration window; and the factor P_{Th} with which the threshold value $t_h[n]$ is decreased. To fix these parameters and to evaluate the performance of the designed algorithm, a sensitivity Se (5) and positive predictivity $+P$ (6) test has been carried out for different configuration parameters:

$$Se = \frac{TP}{TP + FN} \tag{5}$$

$$+P = \frac{TP}{TP + FP} \tag{6}$$

Where TP (True Positive) is the total number of detected R peaks; FN (False Negative) is the total number of non-detected R peaks; and FP (False Positive) is the total number of artefacts or noise classified as R peaks.

After several tests performed over all the databases, it was observed that the parameters had to be different as the sampling frequency F_S changed, in order to achieve higher Se and $+P$. Therefore, the most suitable configuration for parameters N, N_d and P_{Th} has been searched using 10 integer values for N (1 to 10), 10 integer values for N_d (1 to 10), and 21 values for P_{Th} (4.5 to 6.5 in steps of 0.1). Signals from NSRDB (F_S=128Hz) and ADB (F_S=256Hz) databases were selected as searching dataset. Then, the performance was compared by using the minimum value between Se and $+P$ for each combination of these three parameters in the best achieved case. The combination with the best result for the NSRDB was N=3, N_d=2, and P_{Th}=5.4 (F_S=128Hz); whereas in the case of ADB, it was N=6; N_d=5 and P_{Th}=6.1 (F_S=256Hz).

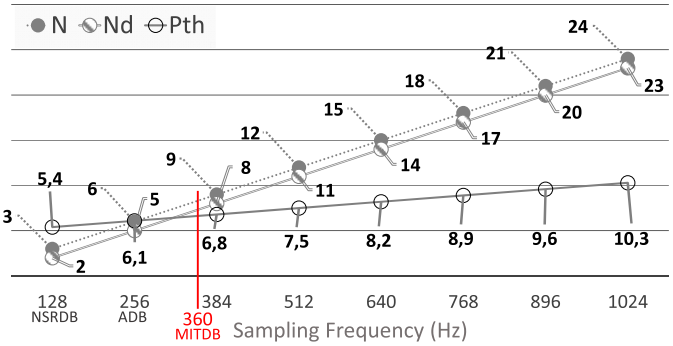

Fig. 7. Estimation of parameters N, N_d and P_{Th} according to the sampling frequency F_S (Hz).

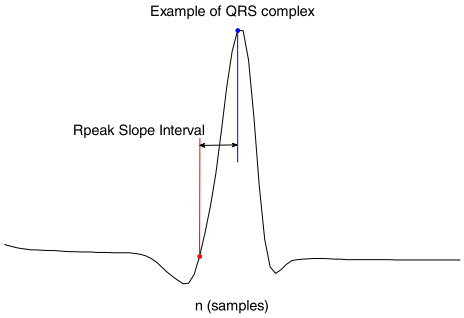

Fig. 8. Definition of R peak slope interval.

Assuming that the parameters N, N_d and P_{Th} have a linear dependency with F_S, these parameters can be defined mathematically by (7), (8) and (9), respectively, by applying a linear interpolation with the sampling frequency F_S:

$$N = round\left(\frac{3 \cdot F_S}{128}\right) \tag{7}$$

$$N_d = round\left(\frac{3 \cdot F_S}{128}\right) - 1 = N - 1 \tag{8}$$

$$P_{Th} = \frac{0.7 \cdot F_S}{128} + 4.7 \tag{9}$$

According to the equations (7)-(9), the parameters for MITDB (F_S=360Hz) are obtained: N=8, N_d=7 and P_{Th}=6.6. Fig. 7 shows the estimated optimal values for the parameters N, N_d, and P_{Th}, in relation with the sampling frequency F_S, for values higher than 128Hz and 256Hz considered in NSRDB and ADB, respectively.

Furthermore, it can be also concluded from the results how the definition of N implies a fixed time of 3/128 seconds (23.4ms), which is approximately the mean time for the R peak slope (Fig. 8), computed for the MITDB, NSRDB and ADB (24.07ms).

V. Results

As was previously stated, the proposed algorithm is characterized for requiring a reduced amount of hardware resources for its implementation, keeping at the same time a similar accuracy level as the most accurate algorithms currently published. Nevertheless, it is not possible to carry

TABLE I
SUMMARY OF THE OBTAINED RESULTS BY THE PROPOSED ALGORITHM WITH THE CONSIDERED DATABASES

Database	*Total peaks*	*TP*	*FN*	*FP*	*Se (%)*	*+P (%)*	*mean (Se, +P)*	*min (Se, +P)*
MITDB	109949	109447	502	289	99.543	99.737	99.640	99.543
NSRDB	192389	192325	64	17	99.967	99.991	99.979	99.967
ADB	197009	195929	1080	1338	99.452	99.322	99.387	99.322
ApneaECG	703750	703053	697	1617	99.901	99.771	99.836	99.771
Fantasia	19445	79436	9	48	99.989	99.940	99.964	99.940
Challenge 2014	72309	72262	47	179	99.935	99.753	99.844	99.753

TABLE II
COMPARISON OF ACCURACY RESULTS FOR SOME PROPOSALS WITH THE MITDB DATABASE

Authors	**Proposed algorithm**		**Results**			**Database**
	Pre-processing stage	Detection stage	*Se*	*+P*	Min	
Pal et al.[24] **(2012)**	EMD	Morphological analysis	99,88	99,96	99,88	21 signals of MITDB
Das et al.[25] **(2013)**	EMD + Wavelet	Dynamic threshold	99,81	99,96	99,81	17 signals of MITDB
Martínez et al.[35] **(2004)**	Wavelet	Set of thresholds	99.80	99.86	99.80	Full MITDB
Christov alg II[20] **(2004)**	Moving averaging filter	3 Dynamic thresholds	99,78	99,78	99,78	Full MITDB
Ghaffari et al. [16]**, (2008)**	Wavelet	Dynamic threshold	99,91	99,72	99,72	Full MITDB
Zidelmal et al.[30] **(2012)**	Wavelet	Dynamic threshold	99,64	99,82	99,64	Full MITDB
Adnane et al.[19] **(2009)**	Differentiation	3 dynamic thresholds	99,77	99,64	99,64	Full MITDB
Phyu et al.[13] **(2009)**	Wavelet	Dynamic threshold	99,63	99,89	99,63	Full MITDB
Nielsen et al.[29] **(2012)**	Wavelet	Dynamic threshold	99,63	99,63	99,63	Full MITDB
Zheng et al.[36] **(2008)**	Wavelet	Dynamic threshold	99,68	99,59	99,59	Full MITDB
Pan & Tomp.[17] **(1985)**	Differentiation	2 dynamic thresholds	99,56	99,76	99,56	Full MITDB
Proposed	**Differentiation**	**Dynamic threshold**	**99,54**	**99,74**	**99,54**	**Full MITDB**
Chen et al.[14] **(2006)**	Wavelet	Dynamic threshold	99,55	99,49	99,49	45 signals of MITDB
Ieong et al.[28] **(2012)**	Wavelet	Dynamic threshold	99,31	99,70	99,31	Full MITDB
Moraes et al.[18] **(2002)**	Differentiation	2 dynamic thresholds	99,22	99,73	99,22	Full MITDB
Laila et al.[27] **(2012)**	Wavelet + Hilbert	Dynamic threshold	96,30	97,83	96,3	Full MITDB

out a detailed comparison about the implementation or the processing time for the different approaches, since most of the previous algorithms are implemented in software. In [6] some algorithms are compared with respect to the computational load, being classified into three categories: low, medium and high computational load. So, for the sake of a hardware and complexity evaluation of the proposed algorithm, we have simulated the well-known Pan & Tompkins algorithm on MATLAB and have tested it with the available data. The simulation has been designed following the same steps as if it was working in real-time, i.e. by using the same sequential processes and operations as if it was running on a portable device.

A. Accuracy Evaluation and Comparison

The use of any QRS complex detector algorithm in medical devices requires a rigorous evaluation of its performance. According to [34], the algorithm should be evaluated by using the sensitivity Se and the positive predictivity +P, already described. Furthermore, in order to achieve comparable results, the evaluation should be carried out on a standard database. In this way, Table I summarizes the results obtained by evaluating the proposed algorithm over the databases described in Section 3. The worst performance is obtained with the Allergy Database, since the ECG signals are corrupted with lots of movement artefacts. Even with this drawback, the averaged sensitivity *Se* is 99.42% and the averaged positive predictivity $+P$ is 99.29%.

Table II shows a comparison among the results obtained by some works previously described. This table is sorted by the minimum value between sensitivity *Se* and positive predictivity $+P$, which is the most common way to compare QRS complex detection algorithms. As can be observed, the accuracy of the proposed algorithm is in the range of the other previous works.

B. Computational Complexity

In order to provide a comparison of the computational complexity of the proposed algorithm, we have also implemented the Pan&Tompkins' algorithm (P&T), likely

TABLE III
PERFORMANCE AND RESOURCE CONSUMPTION FOR THE PAN AND TOMPKINS ALGORITHM AND THE PROPOSED ONE

Algorithm	*Se* (%)	$+P$ (%)	Total resources			Performance			
			Memory cells	Multiplications	Sums	Comp./s	Mult./s	Adds/s	Ops/s
P&T (published)	99.7477	99.5393	-	-	-	-	-	-	-
P&T (implemented by the authors)	99.6564	99.3902	123	18	41	1416	1201	2817	5434
Proposed	99. 5434	99.7366	28	6	5	2163	1107	1205	4475

the most referenced QRS complex detector. For that purpose, we have used the half-hour recordings from the MITDB database and both algorithms have been programmed by using MATLAB to analyze the resources that could be required in the implementation. They have been modeled as they were working in real time (i.e. signals were processed sample by sample, with no use of global buffers assuming the availability of the whole signals), and floating-point representation has been used. The intention has been to test both approaches under the same conditions and constraints. P&T's algorithm has been modeled according to the details given by Pan and Tompkins in [17]. Nevertheless, whereas the pre-processing stage is well detailed, some of the detection stage details are not given (e.g. the initial value of thresholds, their definition of "peak", etc.) and we have made some assumptions, resulting in some differences regarding the obtained results, as is explained next.

The resources required by each algorithm have been estimated. In the case of the proposed algorithm, the number of memory cells depends on the sampling frequency F_S, so the MITDB database is the worst considered case (the higher the sampling rate F_S, the larger number memory units). Table III shows the results about the performance for both algorithms and the resources required for the implementation. Notice the slightly difference between the P&T published results and the performance we have obtained.

Another complexity parameter is the number of operations per second. For that, it is necessary to take into account that P&T's algorithm works at a fixed sampling rate F_S=200Hz (i.e. the MITDB signals were digitized at 360Hz, but they need to be re-sampled at 200Hz for P&T), whereas the proposed one does at F_S=360Hz; working at F_S=200Hz, the proposed algorithm requires almost half of the operations needed at 360Hz. This implies that the proposed algorithm processes more samples per second for the same input signal than the first one. All the comparisons, multiplications and additions have been considered for the 30-minutes long signal from each one of the 48 subjects in the MITDB. Table III shows the results from this analysis: the new algorithm carries out more comparisons than P&T's one, whereas it performs less multiplications and additions.

Whereas the accuracy achieved by both algorithms is quite similar (note the slight difference between the P&T results and the ones obtained here), the computational complexity of the proposed algorithm is lower (less memory cells, less multiplications and less additions), what also implies that the working frequency of the hardware platform can be reduced and so the power consumption. Furthermore, Figures 9-12

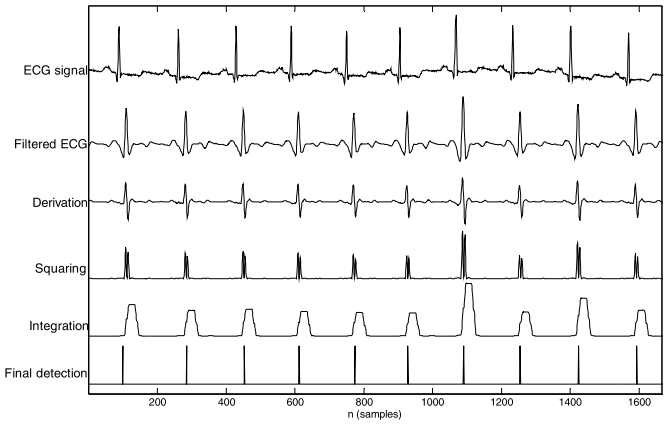

Fig. 9. Pan & Tompkins. QRS complex detection for record no. 100 (*Se*=100%; $+P$=100%; *Fs*=200Hz).

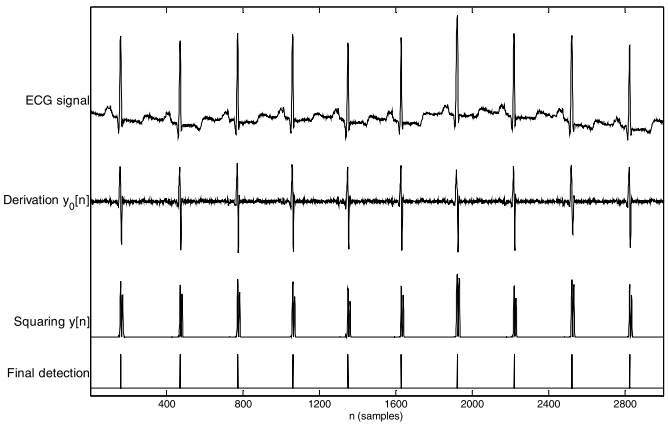

Fig. 10. Proposed algorithm. QRS complex detection for record no. 100 (*Se*=100%; $+P$=100%; *Fs*=360 Hz).

show the detection results for two different ECG segments. The signals are taken from records no. 100 and no. 108 of the MITDB, commonly used for showing the difference between a clear ECG signal (record no. 100) and a very difficult one (record no. 108). These results show again a better detection by the proposed algorithm, especially with noisy recordings. On the other hand, P&T's algorithm integrates the samples to obtain the final trapezoid in which the R peaks are detected. In the cases in which the number of noise peaks is high this integration fuses all of them, so, the resulting signal gets trapezoids produced by noise peaks which are misclassified as R peaks. In other cases, if the QRS complex is too wide (as in the case of the 108m subject, see Fig. 11) the pre-processing

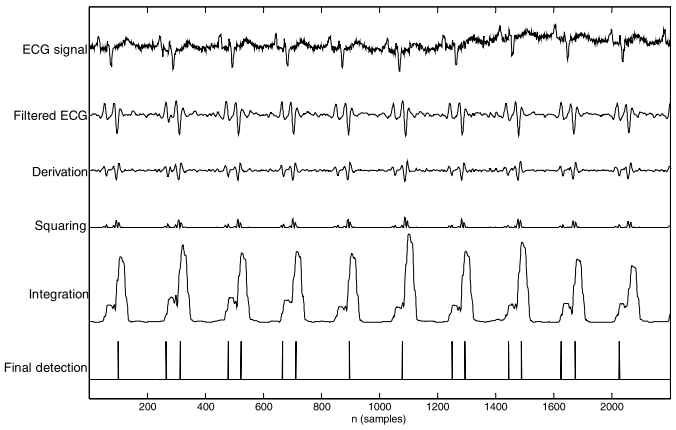

Fig. 11. Pan & Tompkins. QRS complex detection for record no. 108 (Se=98.525%; $+P$=88.083%; Fs=200Hz).

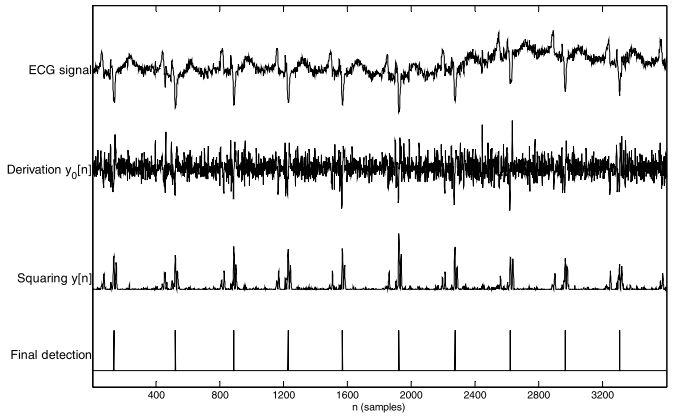

Fig. 12. Proposed algorithm. QRS complex detection for record no. 108 (Se=98.411%; $+P$=95.96%; Fs=360Hz).

TABLE IV
PERFORMANCE AND RESOURCE CONSUMPTION FOR THE PAN AND TOMPKINS' ALGORITHM AND THE PROPOSED ONE

Database	Pan & Tompkins		Proposed	
	Se (%)	$+P$ (%)	*Se* (%)	$+P$ (%)
NSRDB	99.9636	99.9844	99.9667	99.9912
ADB	97.1289	98.9371	99.4518	99.3217
Fantasia	99.9937	99.6022	99.9887	99.9396
Challenge	99.9528	99.7769	99.9350	99.7529
ApneaDB	99.9382	99.6786	99.9010	99.7705

stage generates two trapezoids for each QRS complex. This implies one False Positive (FP) for each wide-QRS-complex. Table IV shows the differences between the results obtained by using the P&T's algorithm and the proposed one over all the databases. Considering the min (Se, $+P$) as a metric for comparing the performances for the best achieved case, the proposal has always a better accuracy except for the Challenge database.

VI. CONCLUSION

A very low-complexity QRS complex detector has been proposed, suitable for wearable systems and for its use in telemedicine or remote monitoring systems. Apart from its low complexity, it achieves sensitivities and specificities about 99.4%, sufficient for most clinical applications. The pre-processing block of the algorithm is mainly based on a differentiation to reduce the low-frequency noises and on an integration to smooth the ECG signal. The detection stage is based on a dynamic threshold, whose value is controlled by a finite state machine, depending on the sampling frequency of the ECG signal.

The behavior of the algorithm has been tested in diverse conditions, through the cases provided by several databases with different pathological recordings, and different artefacts. As one of the databases is considered to be standard, the obtained performance can be easily compared with previous proposals.

The computational time required by the proposed algorithm to detect all the R peaks in the MITDB database has been estimated and compared with the computational time required by the well-known Pan & Tompkins' algorithm in the same conditions. This comparison shows that the proposed algorithm reduces almost 50% the computational time, and it provides better detection with extremely difficult ECG recordings.

REFERENCES

[1] S. Gradl, P. Kugler, C. Lohmuller, and B. Eskofier, "Real-time ECG monitoring and arrhythmia detection using Android-based mobile devices," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug./Sep. 2012, pp. 2452–2455.

[2] S.-H. Liou, Y.-H. Wu, Y.-S. Syu, Y.-L. Gong, H.-C. Chen, and S.-T. Pan, "Real-time remote ECG signal monitor and emergency warning/positioning system on cellular phone," in *Intelligent Information and Database Systems* (Lecture Notes in Computer Science), vol. 7198. Berlin, Germany: Springer-Verlag, 2012, pp. 336–345.

[3] G. G. Berntson *et al.*, "Heart rate variability: Origins, methods, and interpretive caveats," *Psychophysiology*, vol. 34, no. 6, pp. 623–648, Nov. 1997.

[4] M. Bsoul, H. Minn, and L. Tamil, "Apnea MedAssist: Real-time sleep apnea monitor using single-lead ECG," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 3, pp. 416–427, May 2011.

[5] F. Massé, M. Van Bussel, A. Serteyn, J. Arends, and J. Penders, "Miniaturized wireless ECG monitor for real-time detection of epileptic seizures," *ACM Trans. Embed. Comput. Syst.*, vol. 12, no. 4, Jun. 2013, Art. ID 102.

[6] B.-U. Köhler, C. Hennig, and R. Orglmeister, "The principles of software QRS detection," *IEEE Eng. Med. Biol. Mag.*, vol. 21, no. 1, pp. 42–57, Jan./Feb. 2002.

[7] R. Gutiérrez, S. Fernández, J. J. García, and L. Marnane, "Monitoring vital signs and location of patients by using ZigBee wireless sensor networks," in *Proc. IEEE Sensors*, Oct. 2011, pp. 1221–1224.

[8] Y.-L. Zheng *et al.*, "Unobtrusive sensing and wearable devices for health informatics," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 5, pp. 1538–1554, May 2014.

[9] H. L. Kennedy, "The evolution of ambulatory ECG monitoring," *Prog. Cardiovascular Diseases*, vol. 56, no. 2, pp. 127–132, 2013.

[10] H. Xia, I. Asif, and X. Zhao, "Cloud-ECG for real time ECG monitoring and analysis," *Comput. Methods Prog. Biomed.*, vol. 110, no. 3, pp. 253–259, Jun. 2013.

[11] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed sensing for real-time energy-efficient ECG compression on wireless body sensor nodes," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2456–2466, Sep. 2011.

[12] C.-C. Chou, S.-Y. Tseng, E. Chua, Y.-C. Lee, W. Fang, and H.-C. Huang, "Advanced ECG processor with HRV analysis for real-time portable health monitoring," in *Proc. IEEE Int. Conf. Consum. Electron.*, Berlin, Germany, Sep. 2011, pp. 172–175.

[13] M. W. Phyu, Y. Zheng, B. Zhao, L. Xin, and Y. S. Wang, "A real-time ECG QRS detection ASIC based on wavelet multiscale analysis," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2009, pp. 293–296.

[14] S.-W. Chen, H.-C. Chen, and H.-L. Chan, "A real-time QRS detection method based on moving-averaging incorporating with wavelet denoising," *Comput. Methods Prog. Biomed.*, vol. 82, no. 3, pp. 187–195, Jun. 2006.
[15] C. I. Ieong, M. I. Vai, and P. U. Mak, "ECG QRS complex detection with programmable hardware," in *Proc. 30th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2008, pp. 2920–2923.
[16] A. Ghaffari, H. Golbayani, and M. Ghasemi, "A new mathematical based QRS detector using continuous wavelet transform," *Comput. Elect. Eng.*, vol. 34, no. 2, pp. 81–91, Mar. 2008.
[17] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, vol. 32, no. 3, pp. 230–236, Mar. 1985.
[18] J. C. T. B. Moraes, M. M. Freitas, F. N. Vilani, and E. V. Costa, "A QRS complex detection algorithm using electrocardiogram leads," in *Proc. Comput. Cardiol.*, 2002, pp. 205–208.
[19] M. Adnane, Z. Jiang, and S. Choi, "Development of QRS detection algorithm designed for wearable cardiorespiratory system," *Comput. Methods Prog. Biomed.*, vol. 93, no. 1, pp. 20–31, Jan. 2009.
[20] I. I. Christov, "Real time electrocardiogram QRS detection using combined adaptive threshold," *Biomed. Eng. Online*, vol. 3, no. 1, p. 28, Aug. 2004.
[21] R. Rodriguez, A. Mexicano, J. Bila, R. Ponce, S. Cervantes, and A. Martinez, "Hilbert transform and neural networks for identification and modeling of ECG complex," in *Proc. 3rd Int. Conf. Innov. Comput. Technol. (INTECH)*, Aug. 2013, pp. 327–332.
[22] F. Zhang and Y. Lian, "Wavelet and Hilbert transforms based QRS complexes detection algorithm for wearable ECG devices in wireless body sensor networks," in *Proc. IEEE Biomed. Circuits Syst. Conf.*, Nov. 2009, pp. 225–228.
[23] A. Farahabadi, E. Farahabadi, H. Rabbani, and M. P. Mahjoub, "Detection of QRS complex in electrocardiogram signal based on a combination of Hilbert transform, wavelet transform and adaptive thresholding," in *Proc. IEEE-EMBS Int. Conf. Biomed. Health Informat.*, Jan. 2012, pp. 170–173.
[24] S. Pal and M. Mitra, "Empirical mode decomposition based ECG enhancement and QRS detection," *Comput. Biol. Med.*, vol. 42, no. 1, pp. 83–92, Jan. 2012.
[25] M. K. Das, S. Ari, and S. Priyadharsini, "On an algorithm for detection of QRS complexes in noisy electrocardiogram signal," in *Proc. Annu. IEEE India Conf.*, Dec. 2011, pp. 1–5.
[26] S. Kouchaki, A. Dehghani, S. Omranian, and R. Boostani, "ECG-based personal identification using empirical mode decomposition and Hilbert transform," in *Proc. 16th CSI Int. Symp. Artif. Intell. Signal Process. (AISP)*, 2012, pp. 569–573.
[27] I. L. binti Ahmad, M. binti Mohamed, and N. A. binti Ab Ghani, "Development of a concept demonstrator for QRS complex detection using combined algorithms," in *Proc. IEEE EMBS Conf. Biomed. Eng. Sci.*, Dec. 2012, pp. 689–693.
[28] C.-I. Ieong *et al.*, "A 0.83-μW QRS detection processor using quadratic spline wavelet transform for wireless ECG acquisition in 0.35-μm CMOS," *IEEE Trans. Biomed. Circuits Syst.*, vol. 6, no. 6, pp. 586–595, Dec. 2012.
[29] D. B. Nielsen, K. Egstrup, J. Branebjerg, G. B. Andersen, and H. B. D. Sorensen, "Automatic QRS complex detection algorithm designed for a novel wearable, wireless electrocardiogram recording device," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug./Sep. 2012, pp. 2913–2916.
[30] Z. Zidelmal, A. Amirou, M. Adnane, and A. Belouchrani, "QRS detection based on wavelet coefficients," *Comput. Methods Prog. Biomed.*, vol. 107, no. 3, pp. 490–496, Sep. 2012.
[31] A. L. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, Jun. 2000.
[32] *PhysioNet*. [Online]. Available: http://www.physionet.org/physiobank/
[33] Medical Training and Simulation LLC. *EKG Interpretation*. [Online]. Available: http://www.practicalclinicalskills.com/ekg-lesson.aspx?coursecaseorder=2&courseid=301, accessed Mar. 23, 2015.
[34] *ANSI/AAMI EC57:1998/(R)2008: Testing and Reporting Performance Results of Cardiac Rhythm and ST Segment Measurement Algorithms*. AAMI, 2008.
[35] J. P. Martínez, R. Almeida, S. Olmos, A. P. Rocha, and P. Laguna, "A wavelet-based ECG delineator: Evaluation on standard databases.," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 4, pp. 570–581, Apr. 2004.
[36] H. Zheng and J. Wu, "Real-time QRS detection method," in *Proc. 10th Int. Conf. e-Health Netw., Appl. Services (HealthCom)*, 2008, pp. 169–170.

Raquel Gutiérrez-Rivas (S'14) received the B.S. and M.Sc. degrees from the University of Alcalá, Spain, in 2011 and 2012, respectively.

She has been working on different research projects with the Electronics Department, University of Alcalá, since 2009. She is actually working on her Ph.D. thesis. Her research interests include remote monitoring of patients, analysis of physiological signals, and health care systems.

J. Jesús García (M'05) received the Ph.D. (Hons.) degree from the University of Alcalá, Spain, in 2006.

He is currently an Associate Professor of Digital and Analog Electronic with the Electronics Department, University of Alcalá. His research areas are multisensor integration, local positioning systems, and sensory systems for railway safety. He has authored or co-authored over 75 publications in these fields.

William P. Marnane (M'09) received the B.E. degree in electrical engineering from the National University of Ireland, Cork, in 1984, and the Ph.D. degree from the University of Oxford, Oxford, U.K., in 1989.

He has been a Senior Lecturer with the Department of Electrical and Electronic Engineering, National University of Ireland, since 1999. His research interests include biomedical signal processing and digital design for DSP, coding, and cryptography.

Álvaro Hernández (M'06) received the Ph.D. degree from the University of Alcalá, Spain, and Blaise Pascal University, France, in 2003.

He is currently an Associate Professor of Digital Systems and Electronic Design with the Electronics Department, University of Alcalá. His research areas are multisensor integration, electronic systems for mobile robots, and digital and embedded systems.