A Moving Average based Real-time Filtering System for QRS Detection

Klemen Škrlj
Faculty of Computer and Information Science
Univeristy of Ljubljana
ks5379@student.uni-lj.si

Abstract

We implemented a moving average based filtering system based on Chen-Chen detector[1], which can detect QRS complexes in real-time. We then propose improvements to this baseline algorithm. Here we implemented a more robust preprocessing of the original signal as proposed in Pan-Tompkins detector[2]. To reduce the amount of false positive detections, we introduce some additional restrictions in the decision making. All versions of detectors were tested on both LTST[3] and MIT/BIH[4] databases and the porposed improvements have show to have beneficial effect on the results.

1. Introduction

Detection of the QRS complex is a fundamental step when dealing with ECG signals. This is a combination of three graphical deflections (Q, R and S points) visible in the signal which are caused by ventricular depolarization. With detection of such feature we can measure heart rate and with further classification we detect some conditions, which may pose a treat to subject's health.

Because of physiological differences between QRS complexes and various types of noises in the signal, this task has some challenges. Noise is typically a consequence of muscle noise, artifacts due to electrode motion, power line interference and baseline wander. In addition, the T wave which follows the complexes has some times very similar shape and out detector has to be robust enough to not produce false positive detections.

2. Methods

We first implemented baseline algorithem which is described in [1]. This has three main parts. First we filter the input signal with a linear high-pass filter. This is done by substracting signal, filtered with moving average with window size M, from a signal which is delayed by $(M + 1)/2$. Such implementation emphesizes frequency range of desired QRS complex while suppressing low frequences. Those include mainly lower frequency noise, baseline wander and P or T waves. With lower M we risk not emphezing QRS complex enough, while with higher M we may not eliminate as much noise from P or T waves. Empirically we saw that M=7 worked best in our testing. After high pass filtering, the authors continue with non-linear low pass filtering stage. Here the signal is first squared and then a summation with moving window is done. While the authors suggest width of the window to be around 150ms, in our case 60ms worked the best. The third stage of the algorthm is about the decision making process and thresholding. Here a simple equation is presented

$$Threshold = \alpha \cdot \gamma \cdot PEAK + (1 - \alpha) \cdot Threshold \quad (1)$$

where PEAK is a value of current peak that crossed the threshold, α is a value between $0 \leq \alpha \leq 1$ described as forgetting factor and γ is a weighting factor of a PEAK to the threshold correction. Since no threshold initialization is mentioned in the algorithem, we set it to 0.75% of maximum value in the first second of the the signal. In addition, α is set randomly between 0.1 and 0.001 as per authors suggestion.

While the algorithem works in most cases, we made some improvements in the preprocessing steps that were inspired by Pen-Tompkins[2] implementation of their real-time QRS detector. Here we first used a bandpass filter which more accuratly eliminates signal noise outside QRS frequency range (from 5Hz to 15Hz). This is done by first filtering the signal with a low pass filter with transfer characteristic (2)

$$H(z) = \frac{(1 - z^{-6})^2}{(1 - z^{-1})^2} \quad (2)$$

and after that with a high pass filter with transfer characteristic (3)

$$H(z) = \frac{(-1 + 32z^{-16} + z^{-32})}{(1 + z^{-1})} \quad (3)$$

The output of the bandpass filter is then differentiated to provide information about the slope, squared and filtered with moving average filter with a window size of 150ms. After that, we are left with a signal as shown on Figure 1 where desired features are more easily thresholded.

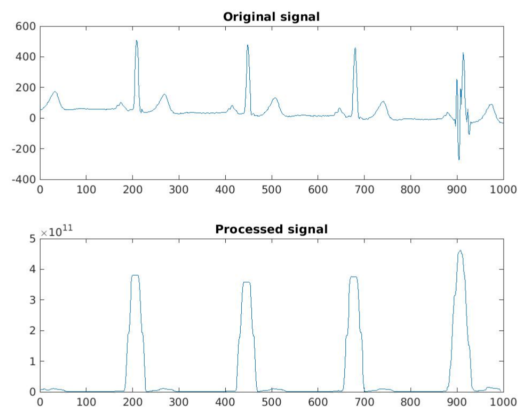

Figure 1. Signal before and after preprocessing steps

In addition to more robust preprocessing, we also improved upon the decision algorithm where we introduce some more constraints. First we set a 200ms refractory period after every valid QRS complex is detected. This means that any local maximums detected inside this window are discarded because physiologically a new QRS couldn't have appeared so fast. And after that we are more careful with detections inside 160ms window after refractory period. Here we have to classify every local maximum as a QRS or a T-wave. To do this we look at an average slope over 10 samples before this detection. If the slope coeficient is less than half of the slope coeficient of previous detection then we classify this detection as a T-wave and we discard it.

3. Results

We tested multiple versions of our algorithm on both LTST and MIT/BIH database to make our results more conclusive. Our results can be seen in Table 1. Here our initial algorithm is labeled as *Original*, *Preprocess* marks algorithm with additional preprocessing, *Min200* implements constraint of a refractory window and *TWave* in the name marks implementation of a T-wave constraint.

Name	LTST		MIT/BIH	
	Se	+P	Se	+P
Original	96.14	88.48	96.84	83.92
Preprocess	**97.20**	82.53	**99.68**	77.44
Min200	96.15	**99.34**	97.25	98.97
PrepMin200	97.12	98.85	99.59	99.58
PrepMin200TWave	96.89	98.91	99.39	**99.62**

Table 1. Results

For evaluating algorithms we choose QRS sensitivity calculated by formula

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

and QRS positive predictivity

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

where TP denotes true positive detections, FP false positive and FN false negative. The highest possible value of both formulas is 100%.

4. Discussion

As we can see from the results, our improvements helped with overall success of the detector. More complex preprocessing steps are ensuring detections of more actual positive QRSs and thus lowering false negatives. While the refractory period of 200ms seems to have greater effect on less false positive detections. This is expected, because some maximums in the sliding window which go over current threshold aren't always true QRS complexes. With this rule we can eliminate such exemples. By the looks of thing our T-wave rule didn't have so large effect on the overall performance but it still has the best overall positive predictivity on MIT/BIH database. Overall we can say that our improvements achive around 97% / 99% sensitivity and 99% / 99% positive predictivity on LTST and MIT/BIH databases respectively. Seems like our decision making could be improved because we have more problems with false negatives then false positives. This could be done with more complex, adjustable thresholding technique.

References

[1] H. Chen and S. Chen, “A moving average based filtering system with its application to real-time QRS detection,” in *Computers in Cardiology, 2003*, IEEE, 2003.

[2] J. Pan and W. J. Tompkins, “A real-time QRS detection algorithm,” *IEEE Transactions on Biomedical Engineering*, vol. BME-32, pp. 230–236, Mar. 1985.

[3] F. Jager, A. Taddei, G. B. Moody, M. Emdin, G. Antolič, R. Dorn, A. Smrdel, C. Marchesi, and R. G. Mark, "Long-term ST database: A reference for the development and evaluation of automated ischaemia detectors and for the study of the dynamics of myocardial ischaemia," *Medical & Biological Engineering & Computing*, vol. 41, pp. 172–182, Mar. 2003.

[4] G. Moody and R. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.

[5] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet," *Circulation*, vol. 101, June 2000.