

QRS Complex Classification Using Simple Distance Measures

Assignment #2

Biometrical signal and image processing 2020/21, Faculty of Computer and Information Science, University of Ljubljana

Žiga Kleine

Abstract—This report presents an implementation of a simple QRS complex classifier, classifying pre-detected heartbeats into normal heartbeats (N), and the abnormal ones, represented only by ventricular ectopic heartbeats (V). It is based on calculating the distance between samples of pre-detected heartbeats. Before classification, I have also eliminated baseline drift using a high-pass Butterworth filter for drift suppression. I used the MATLAB programming framework, which includes many functions useful for digital signal processing. I evaluated the algorithm on the MIT-BIH Arrhythmia Database using the wfdb software package functions bxb and sumstats [1], [2].

I. INTRODUCTION

The problem of QRS complex classification has been tackled many times, usually using some type of feature extraction and comparison between heartbeats, using different techniques of feature extraction and classification. Feature extractors vary from averaging functions to wavelet transform based methods. Classifiers can be simple euclidian or manhattan distance measuring techniques, but in the last years, neural network approaches for classification have been tested [3], [4], [5].

II. METHODOLOGY

The implementation can be split into two tasks. The first task is baseline drift suppression and the second one is feature extraction and classification. But at first, I tried to reduce noise in the recording by working on the signal calculated by averaging all of the channels of the recording.

I solved the baseline drift problem by applying a high-pass Butterworth filter. I set the cutoff frequency to 0.8 Hz. To compensate for the phase shift done by the Butterworth filter, I first filtered the signal, then flipped it around, filtered it again, and lastly flipped it back. This way the phase shift made by the filter was cancelled out.

The classification part basically consists of first selecting the average normal (N) heartbeat to compare it to other heartbeats in the recording. At first I tried to use the sigavg command from the wfdb software package to extract the average normal heartbeat from the first 5 minutes of the recording. But I found out that there can often be baseline drift found in the first few minutes of the recording. So I opted for a different approach. I implemented an algorithm in MATLAB, that loops over the detected normal heartbeats in the first 5 minutes of each recording and then averages all of them. A heartbeat feature vector consists of signal samples from 60 ms before the fiducial point to 100ms after the fiducial point.

The heartbeats were then classified as either normal (N) or as ventricular ectopic heartbeats (V), by calculating distance between our average normal heartbeat and the current heartbeat that we want to classify. I used the distance equation similar to the euclidean distance 1. If the distance between the average beat and the beat of interest was less than 50, the heartbeat was classified as normal, otherwise it was classified as a ventricular ectopic.

$$d2 = \sqrt{\frac{1}{N} * (|x_1 - y_1|^2 + |x_2 - y_2|^2 + \dots + |x_N - y_N|^2)} \quad (1)$$

III. RESULTS

Detections done in MATLAB were appended to .asc type text files. These files were then converted to binary .qrs files that were suitable for the bxb evaluation command from the wfdb toolbox. Detections were compared to modified .atr files, named .fatr, which included annotations just for normal and ventricular ectopic types of heartbeats. Using the command sumstats, results were then gathered in a text file. Then, I calculated the measures of Sensitivity 2, Positive Predictivity 3 and Specificity 4, by gathering True positives, False positives, False negatives and True negatives from the results.txt file by using the following key III:

Nn	true positive
Nv	false negative
Vn	false positive
Vv	true negative

$$Se = \frac{TP}{TP + FN} \quad (2)$$

$$+ P = \frac{TP}{TP + FP} \quad (3)$$

$$Sp = \frac{TN}{TN + FP} \quad (4)$$

The Sensitivity of my implementation is 0.9559, the Positive Predictivity is 0.9891, and the Specificity is 0.8839. These metrics were gathered by running the implemented algorithm on the MIT-BIH Arrhythmia Database, where recordings 107, 109, 111, 118, 124, 207, 214 and 232 were excluded, because there were no normal or ventricular ectopic heartbeats found in these recordings.

IV. DISCUSSION

The results were quite good for such a simple implementation, but could definitely be improved by using more advanced methods of feature extraction and classification, like extraction based on wavelet transform and classification using neural networks.

REFERENCES

- [1] G. B. Moody and R. G. Mark, "The impact of the mit-bih arrhythmia database," *IEEE Engineering in Medicine and Biology Magazine*, vol. 20, no. 3, pp. 45–50, 2001.
- [2] 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, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.
- [3] P. Singh and R. Sharma, "Automatic classification of cardiac signals using lda and qda," *IJRECE*, vol. 2, no. 4, pp. 129–131, 2014.

- [4] S. Chen and H. Chen, "Development of a real-time qrs beat classifier using a nonlinear trimmed moving averaging-based system," in *Computers in Cardiology, 2003*. IEEE, 2003, pp. 577–580.
- [5] B. Abibullaev and H. D. Seo, "A new qrs detection method using wavelets and artificial neural networks," *Journal of medical systems*, vol. 35, no. 4, pp. 683–691, 2011.