A moving average based filtering system with its application to real-time QRS detection

Mateo Kalem, 63170135

Introduction

To fully complete the assignment you have to satisfy the following requirements: The QRS detector has to accept the name of the record (.dat and .hea files of the database) as a parameter. The program has to open this record and detect the QRS complexes in the record. The program has to write an annotation file (.det) in the ASCII format, which contains sample indexes of the fiducial points (FPs) of the detected QRS complexes. The performance of the program has to be evaluated in the sense of sensitivity (Se) and positive predictivity (+P) with regard to manually annotated fiducial points (FPs) of the QRS complexes (.atr files of the database). The results of the performance evaluation together with your discussions and conclusions have to be written into another text (.txt) or document (.pdf) file. Discuss also the weaknesses of the implemented technique and possible improvements. The .det and .txt (or .pdf) files, together with the source of your program (included in a single compressed .zip archive) have to be submitted during uploading the assignment.

Methods

To complete this assignment I used Matlab and the Windows command line, the data which I used in my assignment is from the MIT-BIH database. First my program converts all of the .dat files into the .mat format with the help of the WFDB toolbox. Then I used the wrapper programmes from e-učilnica to construct my main heart detector. The program accepts the name of the record (.mat file) as it's main parameter. Then we filter the input with a high and low-pass filter, with the equations from the article. After the filtering step the program computes the threshold and creates an output file. The outputs are .det files.

Results

To evaluate my program I used sensitivity and positive productivity, which are calculated as follows:

- $Sensitivity = \frac{TP}{\mathrm{TP+FN}}$
- $Positive\ productivity = \frac{\mathrm{TP}}{\mathrm{TP+FP}}$

For window size I chose 10, which follows the guidelines from the referenced article. The sampling frequency of the files is 360 Hz which means that points are generated about every 3 ms. It is important to note that the window size mustn't be too large, since it can capture two heartbeats and consequently detect only one. The other parameters which I used in my detector were:

- M = 5,
- Alpha = 0.05,
- Gamma = 0.15

- Window size for threshold calculation: 180.

With these parameters my detector achieved a sensitivity of 99.11 and positive productivity of 88.33. To calculate the results I used bxb and sumstats from the WFDB toolbox. The results can be found in the file rezultat.txt along with the files eval1.txt and eval2.txt.

Discussion

In conclusion, the detector provides very good results for the MIT-BIH database. The time complexity of the program seems rather short, however it must be noted that the MIT-BIH database is small, containing only 48 ECG records. I believe that for a larger database the time complexity would increase, considering the amount of for loops that my program containts. I had some issues with the bxb function in Matlab, which I fixed by making a batch file to call bxb, otherwise I couldn't evaluate my detector, since bxb caused my program to go into an endless loop. I believe that the positive productivity could be improved in my program if I tested out different paramaters and maybe changed the way I compute the HPF and LPF in the filtering step. Overall, I am satisfied with the provided results.

References

[1] MIT-BIH db, dostopno na: https://physionet.org/content/mitdb/1.0.0/

[2] A moving average based filtering system with its application to real –time QRS detection, Chen, dostopno na: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.591.126&rep=rep1&type=pdf

[3] WFDB toolbox, dostopno na: https://archive.physionet.org/physiotools/matlab/wfdb-app-matlab/