Graphical Abstract (Optional)

To create your abstract, please type over the instructions in the template box below. Fonts or abstract dimensions should not be changed or altered.

Improved Pipelined Wavelet Implementation for Filtering ECG Signals

Aleksandar Milchevski and Marjan Gusev

ELSEVIER

We present a novel algorithm for digital filtering of an electrocardiogram (ECG) signal received by both stationary and non-stationary sensors. The basic idea of digital ECG signal processing is to extract heartbeat frequencies, which are found to be normal in the range between 50 and 200 beats per minute. The extracted heartbeat frequency is found to be irregular if the rate increases or decreases and serves as evidence for a diagnosis of a complex physiological condition.

A lot of noise can be generated from the environment, including the electrical energy supply (50 or 60 *Hz*), breathing, physical movement, muscles etc. We experimented with several digital band pass filters including the finite response filter, Butterworth filter and a filter implementing the digital wavelet transformation. Classical programming of a digital wavelet transformation includes a lot of operations, which increase the algorithm complexity and, therefore, cannot be recommended for smartphones or other mobile devices, due to their limited resources, such as battery life, storage capacity, and processing power. In order to realize a solution that will run on a smartphone and wearable ECG sensor, we faced a challenge to enable a sufficient quality of service and develop an efficient algorithm that will preserve the mobile phone resources.

We implemented a new improved wavelet filter, which uses a circular buffer and, when compared with the classical solution, it reduces the number of memory accesses and instructions that transfer values between data arrays. This algorithm is a highly efficient solution for a smartphone (mobile device) since it decreases the processing time 15-20 times and also saves battery life.

Research Highlights (Required)

To create your highlights, please type the highlights against each `\item` command.

It should be short collection of bullet points that convey the core findings of the article. It should include 3 to 5 bullet points (maximum 85 characters, including spaces, per bullet point.)

- Discrete Wavelet Transformation implementation of a bandpass filter for denoising an ECG signal
- New improved version of a DWT bandpass filter realized with a circular buffer
- Complexity analysis of arithmetic operations and buffer storage requirements
- Time performance analysis and comparison of obtained solution vs existing solutions
-

Pattern Recognition Letters
journal homepage: www.elsevier.com

Improved Pipelined Wavelet Implementation for Filtering ECG Signals

Aleksandar Milchevski[a,**], Marjan Gusev[b]

[a]Ss. Cyril and Methodius University, 1000 Skopje, Macedonia
[b]Ss. Cyril and Methodius University, 1000 Skopje, Macedonia

ABSTRACT

We present a novel algorithm for digital filtering of an electrocardiogram (ECG) signal received by both stationary and non-stationary sensors. The basic idea of digital ECG signal processing is to extract heartbeat frequencies, which are found to be normal in the range between 50 and 200 beats per minute. The extracted heartbeat frequency is found to be irregular if the rate increases or decreases and serves as evidence for a diagnosis of a complex physiological condition. A lot of noise can be generated from the environment, including the electrical energy supply (50 or 60 *Hz*), breathing, physical movement, muscles etc. We experimented with several digital band pass filters including the finite response filter, Butterworth filter and a filter implementing the digital wavelet transformation. Classical programming of a digital wavelet transformation includes a lot of operations, which increase the algorithm complexity and, therefore, cannot be recommended for smartphones or other mobile devices, due to their limited resources, such as battery life, storage capacity, and processing power. In order to realize a solution that will run on a smartphone and wearable ECG sensor, we faced a challenge to enable a sufficient quality of service and develop an efficient algorithm that will preserve the mobile phone resources. We implemented a new improved wavelet filter, which uses a circular buffer and, when compared with the classical solution, it reduces the number of memory accesses and instructions that transfer values between data arrays. This algorithm is a highly efficient solution for a smartphone (mobile device) since it decreases the processing time 15-20 times and also saves battery life.

1. Introduction

The hardware capabilities of smartphones have been growing rapidly and brought a wide range of potential for new innovations. One such innovation is an ECG monitoring application, which is very useful for forecasting heart disease and keeping cardiac patients under surveillance.

Until now, several techniques have been used to filter the signal from noise and extract its essential characteristics. For example, de Lucena et al. (2015) presented a real-time ECG monitoring and processing, which they showed on a smartphone screen. They used three analog filters to filter the noise: a high-pass filter, Notch filter and low-pass filter in their application.

The analog filters, generally, provided less accuracy due to component tolerances. In addition, the analog filter permanently changed the original signal. Had the original signal been sampled, it would have been able to be further processed with various filters.

Preprocessing of the ECG and noise removal is the essential first step in any method using ECG. Pan and Tompkins (1985) introduced one of the most popular methods for QRS detection using a low-pass and high-pass IIR filters to filter the components with a frequency lower than 5 *Hz* or higher than 15 *Hz*. Another technique for de-noising was described by Van Alste et al. (1986). Their technique consisted of using linear phase filter to reduce the baseline wander. They presented results from several FIR filters with different cut-off frequencies and sizes.

Discrete Wavelet Transform (DWT) is another solution for noise removal in ECG signals. The DWT produces a time-scale representation, which is very advantageous for non-stationary signals, such as the ECG. The time-scale representation is a great starting point for feature extraction, which is frequently the follow-up step in this kind of application. Another advan-

**Corresponding author:
e-mail: milchevski@gmail.com (Aleksandar Milchevski)

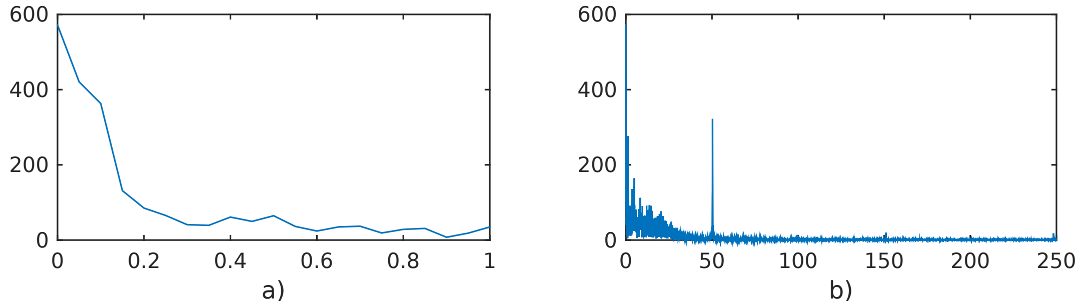

Fig. 1. The frequency domain of an ECG signal: in the range (0,1) Hz (left) and in the range (0,250) Hz (right)

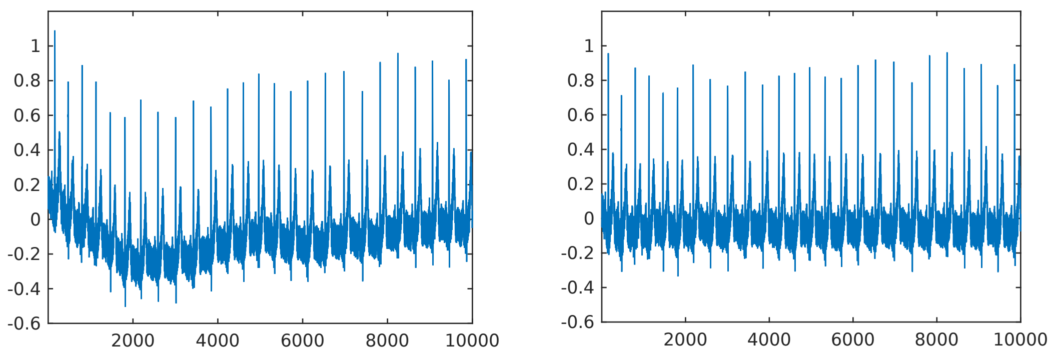

Fig. 2. A baseline drift illustration in the ECG signal (left) and its removal obtained by high-pass filtering (right)

tage of the DWT, compared to conventional filtering methods based on Fourier design, is the sparsity of the time-scale representation. This is important because it allows for noise removal, even in the sub-band of the main components of the ECG.

Von Borries et al. (2006) have presented an example of ECG baseline drift removal using the DWT. The ECG was decomposed into several levels by using the DWT with the biorthogonal 9/7 wavelet. And, the drift was removed by zeroing the obtained coefficients that correspond to the frequency band of the baseline drift, i.e. the lowest frequencies. Two benefits of the use of the DWT were noted. First, the baseline drift was removed without significant distortion to the PQRST complex. Second, it was possible to filter the high-frequency components without distorting the linear phase of the filters. Nemirko and Lugovaya (2005) presented a similar procedure for baseline drift removal by using the DWT with the db8 wavelet. Finally, another benefit of the DWT is that it can be used for ECG delineation. For example, Martínez et al. (2004) used the DWT to present a successful detection of the fiducial points.

We have designed a new implementation of a digital wavelet that filters the noise in the ECG signal by reducing the memory transfer instructions.

2. Analysis of the ECG signal noise removal with Wavelet

Most of the energy of the ECG is situated in the band ($0.3Hz$, $30Hz$) as presented in Fig. 1. The baseline drift and the power supply noise are determined through frequencies below 0.3 Hz and above 30 Hz. Then, a bandpass filtering approach is used to suppresses the recording components outside of this band.

2.1. Noise in ECG signals

The baseline drift is a low-frequency noise, usually present in the band between 0 and 0.5 Hz. It is most commonly caused by breathing, but can also be caused by physical activity made during the recording.

The left part of Fig. 2 presents a typical ECG with a baseline drift, and the right part presents the resulted signal after high-pass filtering. Moreover, one can infer that the recording presented in Fig. 1 has noise because of the amplitude spectrum, which reaches a peak of 50 Hz.

The left part of Fig. 3 shows another type of noise, which is mostly generated by switching power supply and reaches an amplitude spectrum with a frequency of 50/60 Hz. The red line shows the result of applying a low-pass filter, and the blue line the signal input. It is clearly visible that the noise from the power supply has been successfully removed.

Consequentially, by applying both the low-pass and high-pass filters, the above-mentioned noises are eliminated from the ECG recording. A sample of the output of applying the band-pass filter is presented in the right part of Fig. 3.

2.2. Discrete Wavelet Transform

The DWT approach consists of decomposing the input signal into two output signals. The input signal is first filtered through a low pass filter and then the obtained signal is decimated, resulting in the first output signal. The second output signal is derived in a similar way using a high-pass filter, i.e. the input signal is first filtered using the high-pass filter and then the output is down-sampled by a factor of two. Fig. 4 shows the

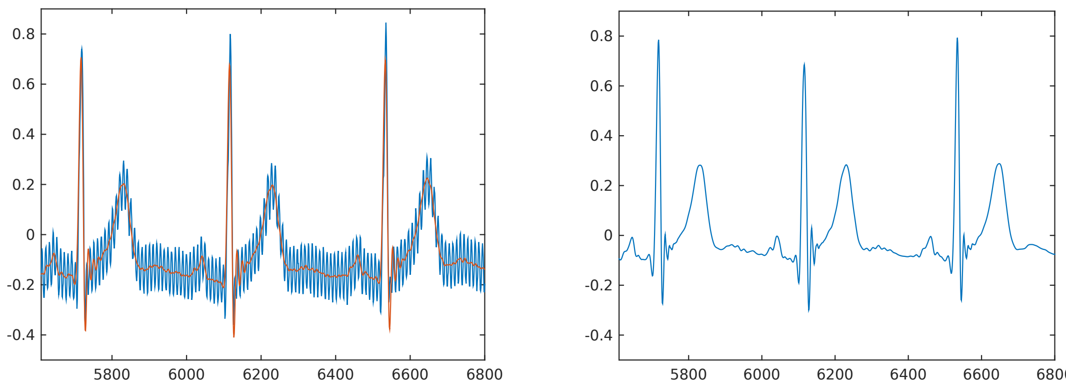

Fig. 3. An ECG signal with power supply noise and low-pass filtering (left) and the result of bandpass filtering (right)

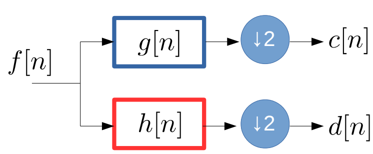

Fig. 4. DWT Analysis

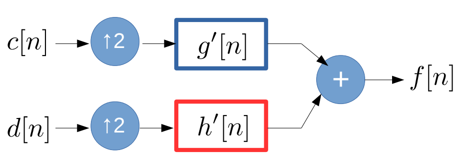

Fig. 5. DWT Synthesis

DWT decomposition, where $h[n]$ and $g[n]$ are the high-pass and low-pass filter coefficients respectively, and $c[n]$ and $d[n]$ the obtained approximation and detail coefficients, respectively, where n is an integer denoting the n-th sample of the signal.

The same procedure is repeated several times by taking the signal with the approximation coefficients as an input into the next step. This leads to a decomposition of a signal into several signals each corresponding to a different sub-band of the spectrum.

Fig. 5 shows the procedure for reconstructing the initial input signal from the obtained approximation and detail coefficients. Both of the signals are first upscaled by a factor of two and then filtered by an appropriate filter. Finally, the sum of the two outputs from the filters is used to reconstruct the initial input signal.

2.3. DWT Filtering of ECG signal noise

The DWT requires three levels of decomposition for high-pass filtering. Namely, if the ECG is sampled with a sampling frequency of Fs = 500 samples/s, and the obtained signal with the wavelet decomposition and the corresponding sub-bands are:

- $c_3[n]$ - third level approximation of the signal, corresponding sub-band: (0, 31.25) Hz
- $d_3[n]$ - third level details of the signal, corresponding sub-band: (31.25, 62.5) Hz
- $d_2[n]$ - second level details of the signal, corresponding sub-band: (62.5, 125) Hz
- $d_1[n]$ - first level details of the signal, corresponding sub-band: (125, 250) Hz

The signal is reconstructed only of the third level approximation of the signal $c_3[n]$, and dismissing the signal details. This leads to an effective filtration of the components with frequencies higher than 31.25 Hz. Then, following the example of Nemirko and Lugovaya (2005), the baseline drift is removed. The approach is similar to the one previously presented for the high-frequency noise, however, the ECG is decomposed using DWT in 9 levels. In the last level, the approximation signal corresponds to the sub-band of (0, 0.49) Hz. All of the signal details are dismissed and a signal is reconstructed using only the last approximation signal. The reconstructed signal is, in fact, the baseline drift and subtracting this signal from the original ECG effectively removes the baseline drift.

3. Design and implementation of DWT filtering

DWT filtering of a signal with a sampling frequency of 500 Hz is a process realized by nine levels of low pass filtering and three levels of high-pass filtering, as presented in Fig. 6.

First, three levels of DWT are calculated and then it is decomposed in two processing parts. The upper processing part calculates 6 more levels and then realizes an inverse DWT on nine levels, while the lower delays the signal and then realizes three levels of inverse DWT, before summing and proceeding to the output.

The naive implementation includes linear buffers for data arrays and data shift for processing of a new sample.

The discrepancy between the number of memory transfer instructions and the real computations was the motivation behind the new pipelined version avoiding as much of the memory transfer instructions, as possible, especially because of the

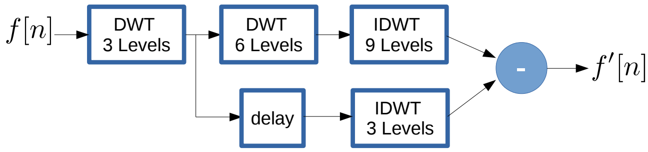

Fig. 6. The block diagram of a DWT filtering of an ECG signal with a sampling frequency of 500 Hz

intended use of a mobile computing device with limited resources.

Our new improved version does not transfer the buffered elements. Instead, we use pointers to the last buffered element, realizing a circular buffer.

Fig. 7 illustrates the difference between the linear and the circular buffer.

4. Time performance and complexity

The following sections present an analysis of buffer storage and processing requirements in order to evaluate the complexity of time performance.

4.1. Buffer storage requirements

The samples from the input signal are processed as they arrive, taking care to minimize the buffering of the intermediary signals as much as possible. The lengths of the buffer of the input signal are equal to the length of the wavelet filter for each DWT level. Additionally, a buffer of the filtered signal is also used to synchronize the obtained baseline drift and the high-pass filtered signal, as shown in Fig. 6.

The size of the buffer c depends on the wavelet length L. On the first level, there is a need to buffer $2 * (M - 1)$ elements, and on each subsequent level L twice as much elements plus the buffer of $M - 1$ elements. This leads to a calculation of $2(\ldots 2(2(M-1)+(M-1))\ldots(M-1))$ for all levels in DWT, which equals to (1).

$$c(L) = (M-1)(2^L - 1) \tag{1}$$

For example, in the naive implementation, the wavelet filter length is $M = 16$, the number of levels for low pass filtering are 9 and 3 for high-pass filtering. Thus, the total buffer is equal to $c = c(9) - c(3) = 15 * 511 - 15 * 7 = 7560$ samples.

4.2. Arithmetic operations

Each DWT requires calculations of M multiplications and M additions, and additional 8 instructions for assigning initial values. Each inverse DWT requires the same number of multiplications and additions as the DWT and adds 4 instructions to assign initial values. These calculations are repeated for each level of DWT.

In addition, each invoking of a DWT or iDWT requires a transfer of M elements of the filter coefficients for each DWT level, and, also, a transfer of all data elements of the buffered signal for its whole length when a new sample is analyzed.

According to this analysis, the number of arithmetic operations for one level of the DWT and IDWT is equal to $2M + 8$ and $2M + 4$ respectively, and M memory transfer instructions. Additionally, the whole buffer is transferred to the left when a new sample is processed, so the number of memory transfers is equal to c. A DWT is invoked $L = 9$ times and iDWT is invoked $L = 9$ times for low pass coefficients plus $L = 3$ times for the high-pass filter coefficients.

This filter is intended to be used by a mobile device that is connected to a wearable sensor, so we need to calculate the total number of all arithmetic operations per processed sample. Since the memory access instructions are the most time critical functions, the processing time is a function of the number of memory accesses.

Three memory accesses are necessary to execute an arithmetic operation based on a multiplication and addition. Whereas, one memory access is necessary in the case of assigning a constant, and two in the case of memory transfer in a typical memory assignment operation.

The total of memory accesses n is given by (2), and will be used as a measure of the processing time.

$$n = 9(3M + 8) + 12(3M + 4) = 63M + 120 \tag{2}$$

However, the number of memory transfer instructions m that move the buffered signal and the filter coefficients one element to the left is presented by (3).

$$m = 9 * 2M + 2c + 12 * 2M = 42M + 2c \tag{3}$$

For example, our first implementation of the DWT uses a wavelet filter with length $M = 16$, leading to 1128 memory accesses for arithmetic instructions and 16464 memory accesses for transfer instructions per each processed sample.

One can notice that the number of memory transfer instructions that prepare the buffered signal to be ready for calculations is huge in comparison to the real number of performed arithmetic instructions. A simple analysis shows that ideally, without analysis of how much processor instructions each procedure call or loop requires, the ratio should be more than 14.5.

4.3. New improved version

The new improved version uses the same number of arithmetic operations and adds 4 more assignment instructions per DWT or iDWT, but requires no memory transfer instructions. Although the total number of arithmetic instructions is increased as presented in (4), the number of memory transfer instructions is 0.

$$n = 9(3M + 12) + 12(3M + 8) = 63M + 204 \tag{4}$$

This analysis confirms that our implementation of a wavelet filter with $M = 16$, leads to 1212 arithmetic operations. A speedup based on an ideal calculation without analyzing processor instructions for procedure calls and realizing loops shows values of 13.5.

Fig. 8 shows the speedup achieved with the improved implementation depending on the length of the input signal. It can be seen that the speedup for all of the tested cases is more than 15 times.

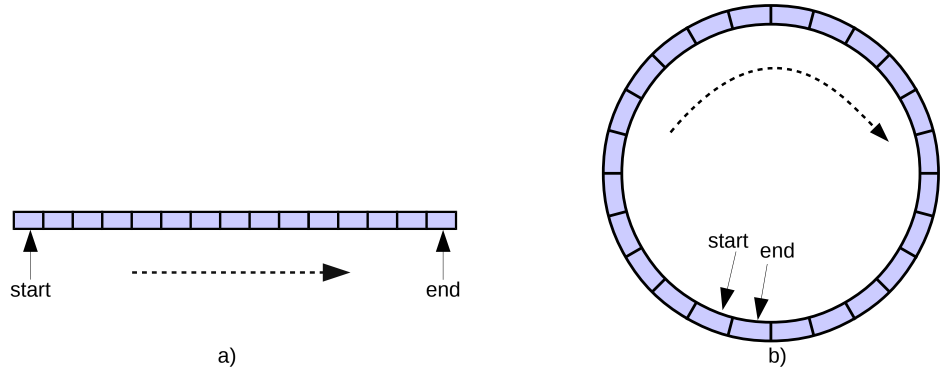

Fig. 7. a) A linear buffer b) A circular buffer

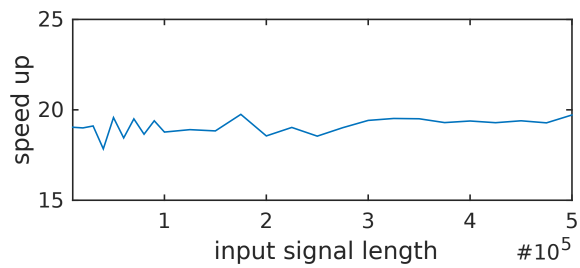

Fig. 8. Theoretical speedup of the improved DWT filtering

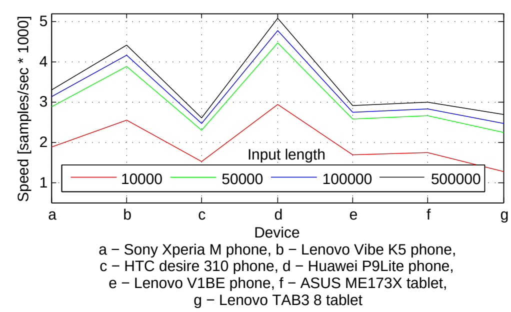

Fig. 9. Performance of conventional DWT filtering of an ECG signal

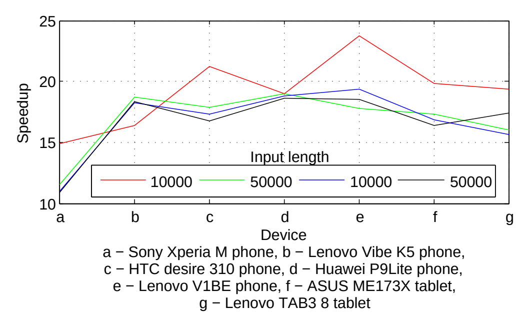

Fig. 10. Experimental speedup of improved DWT filtering

5. Experiment and performance analysis

We have checked the theoretical analysis with real implementations of the algorithm for various Android devices (five smartphones and two tablets). The experiment was realized with ECG input files with lengths of N=10.000, 50.000, 100.000 and 500.000 samples.

Measured execution times of the naive implementation varied from 3.42 sec (Huawei P9 Lite) up to 7,92 sec (Lenovo TAB3 8) for the input of 10.000 samples, and from 98.67 sec (Huawei P9 Lite) up to 192,86 sec (HTC Desire 310) for the input of 500.000 samples. The obtained speed of processed samples per second is presented in Fig. 9.

Measured execution times of the new improved pipelined version with a circular buffer varied from 0.18 sec (Huawei P9 Lite) up to 0.41 sec (Lenovo TAB3 8) for the input of 10.000 samples, and from 5.29 sec (Huawei P9 Lite) up to 13,96 sec (Sony xperia M) for the input of 500.000 samples. The speedup, calculated as a ratio between the corresponding execution times, is presented in Fig. 10.

Since the speedup varies from 10.86 up to 23.76, one can conclude that the architecture influences the performance, although the achieved speedups are within the expected range. We also concluded that the older phones obtained smaller speedup values when compared to the new models.

Another experiment was done in order to compare the execution times of the DWT denoising implementation with the previously designed IIR Butterworth filter. Fig. 11 shows the execution times of the both implementations depending on the length of the input size. It can be seen that the execution times are similar for all of the tested cases.

The presented complexity analysis shows that the proposed algorithm has linear complexity (execution time proportional to the signal input). Furthermore, the scalability of the solution was verified for inputs of up to 500.000 samples. Finally, the measured execution time shows significant improvement, even with the use of the highest optimization options.

6. Related work

Gusev et al. (2016) describe several challenges for developing an m-Health solution with wearable ECG sensors. They conclude that, on one hand, most of the recent research and development tends to use fast processing solutions with accurate feature extraction of ECG signals and, on the other hand, they

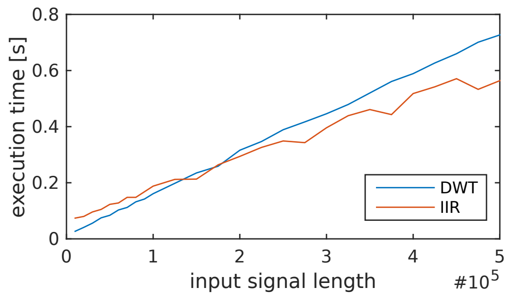

Fig. 11. Execution time of the DWT and IIR Butterworth filter denoising

tend to use solutions that use fewer resources due to limited battery capacity. This is the reason why most of the solutions use FIR/IIR filters, even though DWT is a more accurate method.

The literature overview shows that most of today's published papers that use DWT for ECG signal processing address Matlab services and there are no efforts towards developing new implementation solutions of DWT, or the developed algorithms do not aim to speed up the implementation. In this paper, we conclude that these solutions can be as efficient as the FIR/IIR filters.

Goodfellow et al. (2016) have presented a novel approach to ECG R-peak detection based on DWT as a viable method to assist peak detection with a sensitivity, positive predictivity, and accuracy of over 99.5%, especially for patients with AF (Atrial Fibrillation). A lot of papers address threshold determination in order to obtain a better denoising and all of them include MATLAB implementations for conducting experiments, such as Shemi and Shareena (2016), Zhang et al. (2016), Yadav and Mehra (2016). To our knowledge, none of the existing literature has addressed the problem of realizing a faster DWT implementation.

Addison (2005) has given a comprehensive overview of the use of wavelet transforms in the ECG as a powerful time-frequency analysis and signal processing tool. A lot of solutions for ECG signal denoising using the wavelet transform have been reported Alfaouri and Daqrouq (2008); Kania et al. (2007); Poornachandra (2008); Singh and Tiwari (2006); Saritha et al. (2008), but none has reported the implementation algorithm and its efficiency to be used in a system with limited resources, such as, a smartphone or mobile device. Performance analysis of the signal to noise ratio, mean square error, and other statistics parameters are the only things that have been targeted by researchers.

Soerensen et al. (2010) have presented a comparison of IIR and wavelet filtering for noise reduction of the ECG. They have reported that the wavelet filtering has superior denoising power compared to the IIR filter in the case of high levels of noise. Also, their analysis has shown that, in terms of computational efficiency, the IIR filter implementation operates much faster.

They have also explored the use of the stationary wavelet transform (SWT). The SWT is obtained in a similar way as the DWT, except that the upsamplers and downsamplers are removed. The reported times are 6 seconds for the IIR filters, 88 seconds using the DWT and 272 seconds using the SWT.

In order to compare our implementation, a synthetic ECG was made with the same number of samples. Since the used ECG is a 24-hour recording with a sampling frequency of 200 samples/s, the total number of samples is 17,280,000. The execution times using the created signal are 8 minutes and 23.66 seconds for the naive implementation, 24.94 seconds for the improved DWT implementation and 18.8 seconds for the designed Butterworth filter. The use of compiler optimization for the improved DWT implementation further reduced the execution time to 16.85s.

Kayhan and Ercelebi (2011) have presented another approach for ECG denoising using wavelet transform. In order to evaluate the performance of their denoising method, they have used signals from the MIT-BIH arrhythmia database. The used ECGs were 10 min long and sampled with a frequency of 360 samples/s. The reported execution time using the db8 wavelet was 0.141s.

The execution times using the created signal are 6.39 seconds for the naive implementation, 0.32 seconds for the improved DWT implementation and 0.29 seconds for the designed Butterworth filter. The use of compiler optimization for the improved DWT implementation has further reduced the execution time to 0.22s. However, direct comparison of the methods is not straightforward since Kayhan and Ercelebi (2011) have used 4 levels of decompositions i.e. they are only cleaning the high-frequency noise.

7. Conclusions

In this paper, we present DWT realizations of a bandpass filter, which can be used to digitally process an ECG signal and prepare it for successful feature extraction. The analysis of the wavelet filter characteristics shows good results and the extracted signal contains all of its original features.

Also, a naive implementation of DWT requires a lot of computations in comparison to other filters, and its processing on mobile devices with limited resources can be a time and energy consuming activity.

We have realized a new improved pipelined version using a circular, instead of a linear, buffer that shifts its values for each processed sample of the input signal. The new implementation shows an average speedup of 15 and its processing time is comparable to the FIR and Butterworth IIR filters. We highly recommend the use of this version of the algorithm with mobile devices that have limited resources, such as battery life and processing capabilities.

Gusev et al. (2017) have successfully used the algorithm described here, and initiated the development of a system for early detection and alerting of the onset of a heart attack, thus verifying its usefulness. Moreover, our work is generic and could be a valuable preprocessing step in most of the algorithms where ECG is used, such as the ECG beat classification explained by Martis et al. (2013), and ECG-Based Authentication described by Sufi et al. (2010).

However, future work should be done in order to verify that the presented denoising technique has a positive effect on the accuracy of these algorithms.

References

Addison, P.S., 2005. Wavelet transforms and the ECG: a review. Physiological measurement 26, R155.

Alfaouri, M., Daqrouq, K., 2008. ECG signal denoising by wavelet transform thresholding. American Journal of applied sciences 5, 276–281.

Goodfellow, J., Escalona, O.J., Kodoth, V., Manoharan, G., Bosnjak, A., 2016. Denoising and automated r-peak detection in the ecg using discrete wavelet transform, in: Computing in Cardiology Conference (CinC), 2016, IEEE. pp. 1045–1048.

Gusev, M., Stojmenski, A., Chorbev, I., 2016. Challenges for development of an ECG m-health solution. Journal of Emerging Research and Solutions in ICT 1, 25–38.

Gusev, M., Ristovski, A., Guseva, A., 2017. Pattern recognition of a digital ECG, in: ICT Innovations 2016, Advances in Intelligent and Soft Computing, (ed. A. Kulakov and G. Stojanov), Springer Verlag, Berlin Heidelberg, in press.

Kania, M., Fereniec, M., Maniewski, R., 2007. Wavelet denoising for multi-lead high resolution ECG signals. Measurement science review 7, 30–33.

Kayhan, S., Ercelebi, E., 2011. ECG denoising on bivariate shrinkage function exploiting interscale dependency of wavelet coefficients. Turkish Journal of Electrical Engineering & Computer Sciences 19, 495–511.

de Lucena, S.E., Sampaio, D.J., Mall, B., Meyer, M., Burkart, M.A., Keller, F.V., 2015. ECG monitoring using Android mobile phone and Bluetooth, in: Instrumentation and Measurement Technology Conference (I2MTC), 2015 IEEE International, IEEE. pp. 1976–1980.

Martínez, J.P., Almeida, R., Olmos, S., Rocha, A.P., Laguna, P., 2004. A wavelet-based ECG delineator: evaluation on standard databases. Biomedical Engineering, IEEE Transactions on 51, 570–581.

Martis, R.J., Acharya, U.R., Min, L.C., 2013. ECG beat classification using pca, lda, ica and discrete wavelet transform. Biomedical Signal Processing and Control 8, 437–448.

Nemirko, A., Lugovaya, T., 2005. Biometric human identification based on electrocardiogram, MAKS Press, Moscow. pp. 387–390.

Pan, J., Tompkins, W.J., 1985. A real-time QRS detection algorithm. Biomedical Engineering, IEEE Transactions on , 230–236.

Poornachandra, S., 2008. Wavelet-based denoising using subband dependent threshold for ECG signals. Digital signal processing 18, 49–55.

Saritha, C., Sukanya, V., Murthy, Y.N., 2008. ECG signal analysis using wavelet transforms. Bulg. J. Phys 35, 68–77.

Shemi, P., Shareena, E., 2016. Analysis of ecg signal denoising using discrete wavelet transform, in: Engineering and Technology (ICETECH), 2016 IEEE International Conference on, IEEE. pp. 713–718.

Singh, B.N., Tiwari, A.K., 2006. Optimal selection of wavelet basis function applied to ECG signal denoising. Digital Signal Processing 16, 275–287.

Soerensen, J.S., Johannesen, L., Grove, U.S.L., Lundhus, K., Couderc, J.P., Graff, C., 2010. A comparison of IIR and wavelet filtering for noise reduction of the ECG, in: Computing in Cardiology, 2010, IEEE. pp. 489–492.

Sufi, F., Khalil, I., Hu, J., 2010. ECG-based authentication, in: Handbook of information and communication security. Springer, pp. 309–331.

Van Alste, J., Van Eck, W., Herrmann, O., 1986. ECG baseline wander reduction using linear phase filters. Computers and Biomedical Research 19, 417–427.

Von Borries, R., Pierluissi, J., Nazeran, H., 2006. Wavelet transform-based ECG baseline drift removal for body surface potential mapping, in: Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the, IEEE. pp. 3891–3894.

Yadav, T., Mehra, R., 2016. Denoising and snr improvement of ecg signals using wavelet based techniques, in: Next Generation Computing Technologies (NGCT), 2016 2nd International Conference on, IEEE. pp. 678–682.

Zhang, J., Lin, J.L., Li, X.L., Wang, W.Q., 2016. Ecg signals denoising method based on improved wavelet threshold algorithm, in: Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2016 IEEE, IEEE. pp. 1779–1784.