Real-Time PPG-Based HRV Implementation Using Deep Learning and Simulink

Filipa Cardoso, Arnaldo Batista, Valentina Vassilenko, Manuel Ortigueira

Real-Time PPG-Based HRV Implementation Using Deep Learning and Simulink

Filipa Esgalhado[1,2], Arnaldo Batista[3], Valentina Vassilenko[1,2] and Manuel Ortigueira[3]

[1] Laboratory of Instrumentation, Biomedical Engineering and Radiation Physics (LIBPHYS), NOVA School of Science and Technology - NOVA University Lisbon, 2829-516 Caparica, Portugal

[2] NMT, S.A., Parque Tecnológico de Cantanhede, Núcleo 04, Lote 3, 3060 - 197 Cantanhede, Portugal

[3] UNINOVA CTS, NOVA School of Science and Technology - NOVA University Lisbon, 2829-516 Caparica, Portugal

Abstract. The Heart Rate Variability (HRV) signal computation relies on fiducial points typically obtained from the electrocardiogram (ECG) or the photoplethysmogram (PPG). Generally, these fiducial points correspond to the peaks of the ECG or PPG. Consequently, the HRV quality depends on the fiducial points detection accuracy. In a previous work, this subject has been addressed using Long Short-Term Memory (LSTM) Deep Learning algorithms for PPG segmentation, from which peak detection can be achieved. In the herein presented work, a *Simulink*® implementation of the LSTM algorithm is obtained for real-time PPG peak detection. HRV and outlier removal blocks are also implemented. The obtained code can be used to be embedded in hardware systems for real-time PPG acquisition and HRV visualization. A Root Mean Square Error (RMSE) mean of 0.0439±0.0175 seconds was obtained, and no significant differences (p-value<0.05) were found between the ground truth and the real-time implementation.

Keywords: PPG; HRV; Real-Time; *Simulink*®.

1 Introduction

Cardiovascular diseases (CVD) are the leading cause of death worldwide and are a major public health issue [1]. Therefore, it is essential to adopt preventive strategies for CVD's early detection.

The Heart Rate Variability (HRV) represents the time interval between successive heart beats and can be used to assess the status of the autonomic nervous system [2]. This marker reflects the balance between the parasympathetic and sympathetic systems and has been used in the prediction of cardiovascular outcomes [3]. Several time, frequency and non-linear features can be extracted from the HRV, such as the Standard Deviation of Normal-Normal intervals (SDNN), low frequency peak and sample entropy [4]. Some HRV features, such as the SDNN, have been deemed as useful to predict adverse cardiac outcomes [5]. The HRV can be calculated by determining the systolic peak of the photoplethysmogram (PPG) or the R peak of the electrocardiogram (ECG). The PPG signal measures blood volume variations in the microvascular system through an infrared light sensor placed on the skin surface, thus making this a non-invasive technique. Peak detection is an essential step in the determination of the HRV, since erroneously detected or unaccounted peaks lead to errors in the inter-beat time intervals, which affects the features accuracy [6]. The state-of-the-art accounts herein presented refer only to real time systems implementation. The commonly used platforms include Simulink® and LabView.

Real time systems have been developed to process ECG and PPG signals. Tanji et al. [7] developed an ECG noise rejection system in Simulink® for the attenuation of the electromagnetic interference. Tejaswi et al. [8] implemented an Recursive Least Square (RLS) filter in Simulink®, to reduce noise in the ECG signal. Bhogeshwar et al. [9] implemented different filters to denoise the ECG. Mukherjea et al. [11] created a model to generate synthetic PPG waveforms. The developing platform was the Simulink® in both cases. Shiraishi et al. [10] analysed, in real time, the HRV variations during exercise, based on a 12-lead ECG. Bagha et al. [12] developed in LabView, a real time analysis of blood-oxygen saturation levels (SpO_2).

The goal of this work was to obtain an HRV estimation from the PPG signal by developing a Simulink® based real-time system with a machine learning module. The innovative aspect of the herein presented work consists on the use of a machine learning algorithm working on PPG signals under the general framework of a real time prototyping platform. Despite the offline nature of the PPG data used in this work, the implementations will be referred as being real time.

2 Methods

A real time peak detection and HRV estimation system was developed in Simulink®. To test and debug the developed system, previously recorded PPG signals with a sampling rate of 50 Hz were used as the input. Figure 1 represents an overview of the developed Simulink® model which is comprised by three main blocks: *Deep Learning*, *Find Peaks* and *HRV Computation*. A detailed explanation of each block is presented in the respective sub-section.

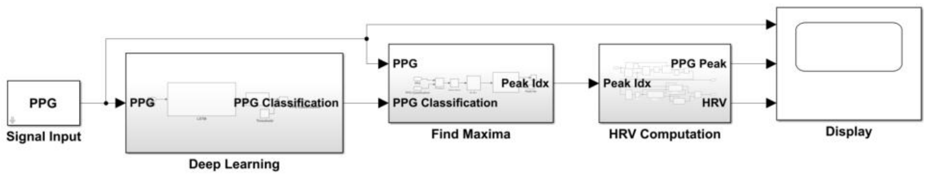

Fig. 1. Real-time HRV estimation system developed in Simulink®. This system is comprised by three main blocks: *Deep Learning, Find Peaks and HRV Computation*.

2.1 Deep Learning Block

The first block of the prototype is based on a Deep Learning (DL) model that was implemented and tested in the authors' previous work [13], where all the system implementation is detailed and explained. This model was optimized and tested for PPG signals. Figure 2 outlines the implementation steps of the DL model.

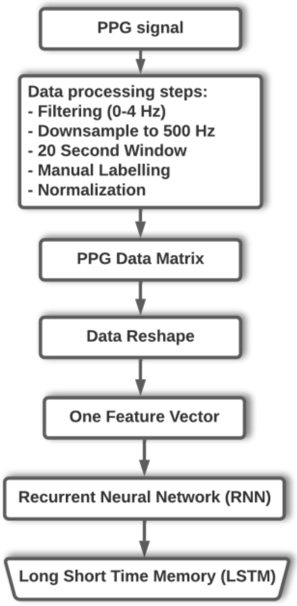

Fig. 2. Flowchart of the implementation steps of the LSTM model as described in [13].

The single layer Long Short-Term Memory (LSTM) model with 200 neurons that was trained in MATLAB and saved in a *mat* format file was subsequently uploaded to the Simulink® based system using the native *Predict* block. The goal of the LSTM network was to classify each data point in two categories: noise or signal. The output of the model is a probability value. Therefore, it was necessary to establish a threshold level for the categories' separation. The LSTM model architecture and the threshold of 0.515 used in the herein presented work were based on the results in [13]. The output of this block is a logic array, where the zero and one correspond to the noise and signal categories, respectively. Figure 3 shows the outline of the *Deep Learning* block.

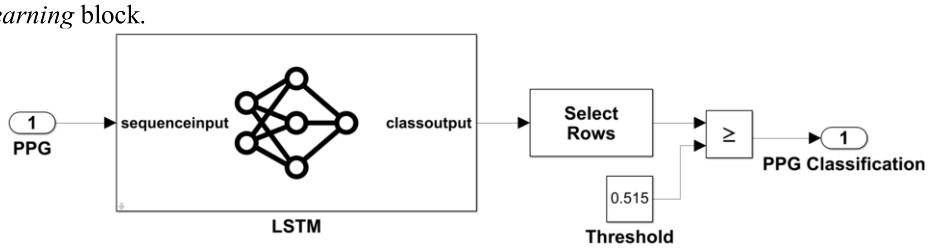

Fig. 3. *Deep Learning* block, with a PPG input, LSTM model and PPG classification.

2.2 Find Peaks Block

After the data classification performed by the Deep Learning block, each data point was classed into two categories: noise and signal. The first step of the *Find Peaks* block is a multiplication of the inputs. As shown in Figure 4, input 1 is the PPG signal, and input 2 is the output of the Deep Learning block. The format of this output is a logic array where each datapoint is defined by its category. This method ensures that the signal points classified as noise by the deep learning model will not be wrongly identified as peaks. Noise samples will be categorized with zero value in the PPG signal. This step enhances system robustness to noise artifacts. After this step, the peak detection algorithm is implemented. Since a buffer block with three samples using a two-sample overlap was applied for peak detection, it is necessary to have at least three signal samples. The buffer block creates a signal frame which is the input for the native Simulink® *Find Maxima* block, used to obtain the signal peaks. The output of the *Find Peaks* block is the peak vector index. Figure 4 presents in detail the *Find Peaks* block.

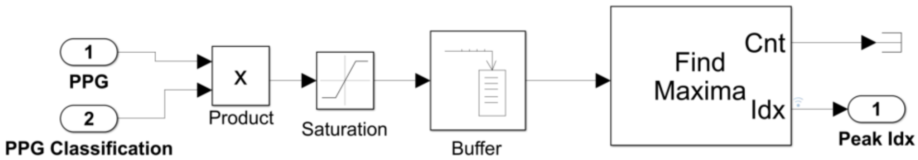

Fig. 4. *Find Peaks* block with two inputs: the PPG signal and its deep learning classification

This block output was the peak index location.

The blocks depicted in Figure 4 reads as follows:

- A product block implements the multiplication between the PPG signal and the categorical variable that is the output of the LSTM classifier.
- A saturation block ensures that the excursion of the product signal stands from a value close to zero to the PPG peak.
- The buffer block produces a frame of 3 samples, which will be a requirement of the next block, the *Find Peaks*.

2.3 HRV Computation Block

The upper line of the HRV block shown in Figure 5 was methodological implemented as in [14]. An outlier detector was included, represented in the *If* blocks of the lower line of the Figure 5, to increase HRV estimation accuracy. The output of the *Find Peaks* block, the peak index, is the input of this block. The blocks in Figure 5 are described as follows:

- A native *Unbuffer* block was applied to return the signal to its original sample-by-sample format.
- The unbuffered signal is input to an integrator and accumulator processor based on the backward Euler method, with one zero and one pole. This cumulative sum operation is an essential part of the HRV estimation.

- The signal is then integrated and delayed. A native *Sample and Hold* block was implemented in order to find the sample distance between consecutive peaks [14]. This block is triggered by the signal transitions in the *Peak Idx* input.
- After determining, in number of samples, the distance between peaks it is necessary to convert this value to seconds, in order to obtain the HRV. To achieve this, the distance is divided by the 50 Hz sampling frequency.
- To minimize the possibility of peak detection errors and increase the robustness of the developed system, an *IF cycle* based on the HRV values was introduced. A threshold of 0.3 seconds is applied. The *IF cycle* evaluated if a difference above 0.3 seconds is present in the HRV vector. If this difference is detected the HRV maintained its previous value. The *Merge* block combines the result of the *If* and *Else* blocks on a single output. The outputs of the *HRV Computation* block are the PPG peak location, output 1, and the HRV estimation, output 2.

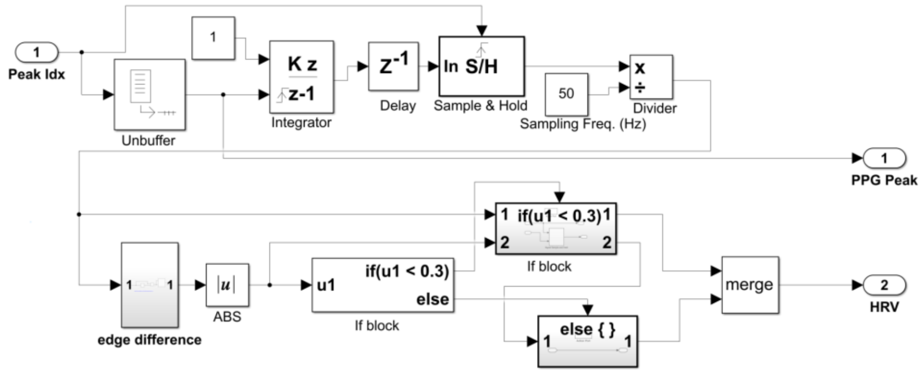

Fig. 5. *HRV Computation* block. From the peak index input, a sample distance between peaks is determined from which the HRV can be obtained. More details can be found in the text.

3 Results

Figure 6 shows the output of the developed system for a 10 second PPG input. This is depicted in the upper plot. The second plot shows the PPG peak location determined with the *Find Peaks* block. The third plot presents the real time HRV estimation, obtained from the time interval between successive peaks. In Figure 7 a PPG signal with 160 seconds is represented, where it is possible to observe the natural signal oscillations as a result of the sympathetic and parasympathetic system intervention.

To evaluate the robustness of the designed system, a comparison of HRV vectors was performed. Twenty 10 second length PPG signals were analysed. To establish the ground truth, the PPG peaks were marked by expert observation and the HRV was calculated. This HRV ground truth was compared to the HRV extracted from the Simulink®. To compare the performance of system, the Root Mean Square Error (RMSE) and a T-test were performed. A 0.05 significance value was selected for the

T-test. The results presented in Table 1 show that the real time HRV estimation adequately represents the ground truth with 0.0439 seconds and 0.0175 seconds of RMSE mean and standard deviation, respectively. The T-test results revealed no significant differences between the evaluated HRV vectors.

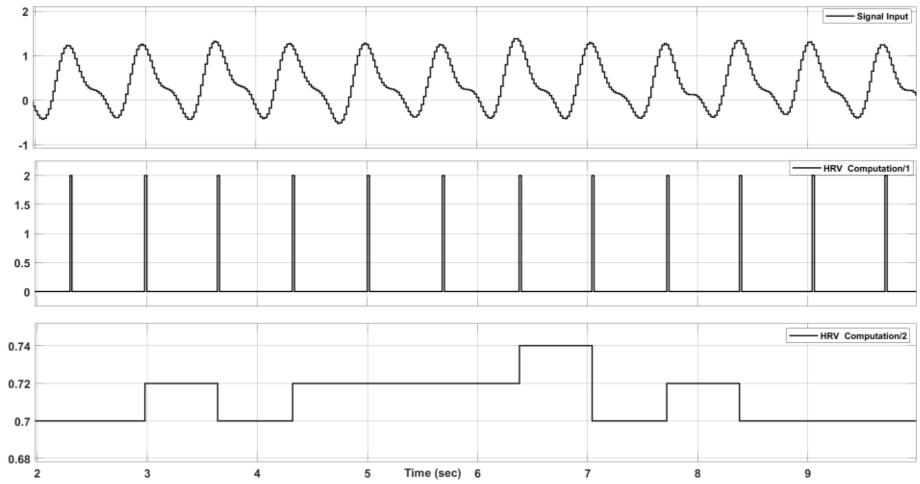

Fig. 6. Ten seconds display output of the Simulink® system. The PPG input signal, peak detection and HRV estimation are represented in the first, second and third plot, respectively.

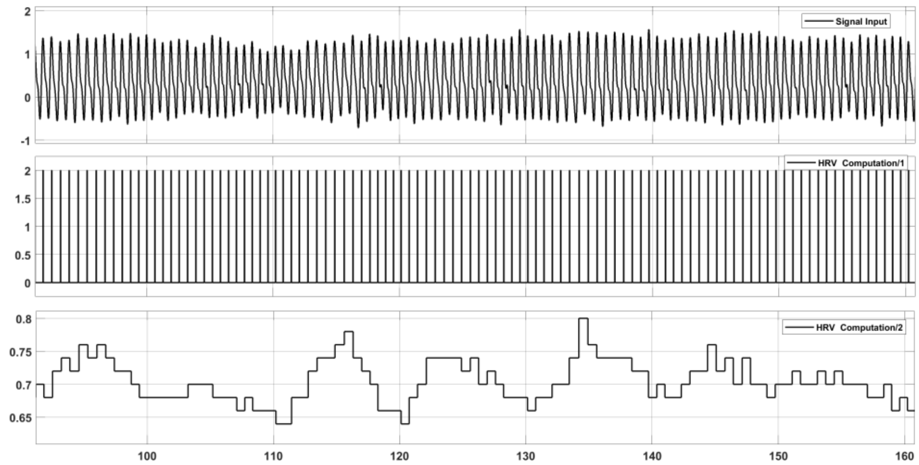

Fig. 7. Display output of the Simulink® system of a PPG signal of 160 seconds. The PPG input signal, peak detection and HRV estimation are represented in the first, second and third plot, respectively.

Table 1. RMSE and T-test results from comparison of HRV vector of gold standard vs. Simulink®.

PPG Signal	RMSE (seconds)	p-value
1	0.040	8.900e-12
2	0.040	1.419e-12
3	0.040	3.742e-12
4	0.041	4.996e-05
5	0.040	4.139e-12
6	0.041	4.995e-05
7	0.040	4.139e-12
8	0.040	1.211e-12
9	0.040	2.358e-13
10	0.047	2.428e-04
11	0.024	1.609e-06
12	0.040	1.171e-12
13	0.040	1.765e-12
14	0.040	6.449e-12
15	0.040	4.522e-12
16	0.040	4.427e-12
17	0.116	9.093e-04
18	0.048	8.488e-06
19	0.040	3.244e-12
20	0.040	1.199e-12

4 Discussion and Conclusion

Nowadays, wearable devices with incorporated PPG sensors are widely available. Smartwatches are an example of such devices that can be used to record biomedical signals, increasing the need for robust real-time algorithms that can compute different features. It is expected that real time deep learning implementation will become popular in wearable devices due to its robustness and increased accuracy. The herein developed system represents an innovation in this respect since it incorporates a LSTM machine learning module.

The results of the herein presented work are a follow up of the authors' theoretical and practical previous implementation in offline platforms. It is often challenging to implement offline algorithms in real time systems due to non-casual processing issues or latency problems. In future work, different deep learning models previously studied will be implemented and tested in the Simulink® system. The produced code could be embedded in hardware platforms such as Android based systems. A simple evaluation of the processing speed of the herein presented model shows that the code can be implemented in the average current Android system without speed or overload issues. Regarding the PPG signal acquisition system on the patient side, it is within the authors' future work to use an Arduino acquisition system with a wi-fi link to the Android system. The herein used 50 Hz PPG sampling frequency is well within the Arduino capabilities.

Acknowledgments. This work was funded and supported by the Fundação para a Ciência e Tecnologia (FCT, Portugal) and NMT, S.A in the scope of the PhD grant PD/BDE/150312/2019.

References

[1] F. B. Ahmad and R. N. Anderson, "The Leading Causes of Death in the US for 2020," *JAMA*, vol. 325, no. 18, p. 1829, May 2021, doi: 10.1001/jama.2021.5469.

[2] Guidelines, T. N. American, Guidelines, and Task Force of the European Society of Cardiology the North American Society of Pacing Electrophysiology, "Guidelines Heart rate variability," *Eur. Heart J.*, vol. 17, pp. 354–381, 1996, doi: 10.1161/01.CIR.93.5.1043.

[3] N. Singh, K. J. Moneghetti, J. W. Christle, D. Hadley, D. Plews, and V. Froelicher, "Heart Rate Variability: An Old Metric with New Meaning in the Era of using mHealth Technologies for Health and Exercise Training Guidance. Part One: Physiology and Methods," *Arrhythmia Electrophysiol. Rev.*, vol. 7, no. 3, p. 193, 2018, doi: 10.15420/aer.2018.27.2.

[4] F. Shaffer and J. P. Ginsberg, "An Overview of Heart Rate Variability Metrics and Norms," *Front. Public Heal.*, vol. 5, no. September, pp. 1–17, 2017, doi: 10.3389/fpubh.2017.00258.

[5] F. Buccelletti *et al.*, "Heart rate variability and myocardial infarction: Systematic literature review and metanalysis," *Eur. Rev. Med. Pharmacol. Sci.*, vol. 13, no. 4, pp. 299–307, 2009, [Online]. Available: https://www.researchgate.net/publication/26754890.

[6] D. C. Sheridan, R. Dehart, A. Lin, M. Sabbaj, and S. D. Baker, "Heart Rate Variability Analysis: How Much Artifact Can We Remove?," *Psychiatry Investig.*, vol. 17, no. 9, pp. 960–965, Sep. 2020, doi: 10.30773/pi.2020.0168.

[7] A. K. Tanji, M. A. G. de Brito, M. G. Alves, R. C. Garcia, G.-L. Chen, and N. R. N. Ama, "Improved Noise Cancelling Algorithm for Electrocardiogram Based on Moving Average Adaptive Filter," *Electronics*, vol. 10, no. 19, p. 2366, Sep. 2021, doi: 10.3390/electronics10192366.

[8] V. Tejaswi, A. Surendar, and N. Srikanta, "Simulink implementation of RLS algorithm for resilient artefacts removal in ECG signal," *Int. J. Adv. Intell. Paradig.*, vol. 16, no. 3/4, p. 324, 2020, doi: 10.1504/IJAIP.2020.107529.

[9] S. S. Bhogeshwar, M. K. Soni, and D. Bansal, "Design of Simulink Model to denoise ECG signal using various IIR & FIR filters," in *2014 International Conference on Reliability Optimization and Information Technology (ICROIT)*, Feb. 2014, pp. 477–483, doi: 10.1109/ICROIT.2014.6798370.

[10] Y. Shiraishi *et al.*, "Real-Time Analysis of the Heart Rate Variability During Incremental Exercise for the Detection of the Ventilatory Threshold," *J. Am. Heart Assoc.*, vol. 7, no. 1, Jan. 2018, doi: 10.1161/JAHA.117.006612.

[11] A. Mukherjea, P. Chaudhury, A. Karkun, S. Ghosh, and S. Bhowmick, "Synthesis of PPG Waveform Using PSPICE and Simulink Model," in *2019 Devices for Integrated Circuit (DevIC)*, Mar. 2019, pp. 428–432, doi: 10.1109/DEVIC.2019.8783684.

[12] S. Bagha and L. Shaw, "A Real Time Analysis of PPG Signal for Measurement of SpO2 and Pulse Rate," *Int. J. Comput. Appl.*, vol. 36, 2011, doi: 10.5120/4537-6461.

[13] F. Esgalhado, B. Fernandes, V. Vassilenko, A. Batista, and S. Russo, "The Application of Deep Learning Algorithms for PPG Signal Processing and Classification," *Computers*, vol. 10, no. 12, p. 158, 2021, doi: 10.3390/computers10120158.

[14] T. Lukáč and O. Ondráček, "Using Simulink and Matlab for Real-Time ECG Signal Processing," *Conf. MATLAB*, 2012.