END-TO-END PHOTOPLETHYSMOGRAPHY (PPG) BASED BIOMETRIC AUTHENTICATION BY USING CONVOLUTIONAL NEURAL NETWORKS

Jordi Luque[1], *Guillem Cortès*[1], *Carlos Segura* [1],
Alexandre Maravilla[2], *Javier Esteban*[2], *Joan Fabregat*[2]

[1] Telefónica Research Edificio Telefónica-Diagonal 00, Barcelona, Spain
[2] Telefónica de España, Spain

ABSTRACT

Whilst research efforts have traditionally focused on Electrocardiographic (ECG) signals and handcrafted features as potential biometric traits, few works have explored systems based on the raw photoplethysmogram (PPG) signal. This work proposes an end-to-end architecture to offer biometric authentication using PPG biosensors through Convolutional Networks. We provide an evaluation of the performance of our approach in two different databases: Troika and PulseID, the latter a publicly available database specifically collected by the authors for such a purpose. Our verification approach through convolutional network based models and using raw PPG signals appears to be viable in current monitoring procedures within e-health and fitness environments showing a remarkable potential as a biometry. The approach tested on a verification fashion, on trials lasting one second, achieved an AUC of 78.2% and 83.2%, averaged among target subjects, on PulseID and Troika datasets respectively. Our experimental results on previous small datasets support the usefulness of PPG extracted biomarkers as viable traits for multi-biometric or standalone biometrics. Furthermore, the approach results in a low input throughput and complexity that allows for a continuous authentication in real-world scenarios. Nevertheless, the reported experiments also suggest that further research is necessary to account for and understand sources of variability found in some subjects.

***Index Terms*—** photoplethysmogram signal, ppg, biometric authentication, biometric verification, convolutional neural networks

1. INTRODUCTION

User authentication based on monitoring the heart signal has raised the interest of the research community due to the increasingly popularity of wereable biosensors. Wrist-type photoplethysmographic (PPG) sensors have become a standard in health care and fitness applications owing to their capabilities for low cost and long term screening. Despite the fact that PPG signals can be easily obtained from the finger or by wrist-type wearables and smart-watches, it arises several questions about its potential and viability as a biometric trait, e.g. due to motion artifacts, as well as around the selection of appropriate biomarkers.

The PPG sensor is a non-invasive electro-optical method [1] that provides the PPG signal as illuminance variations measured by a photo-detector. Usually, a source of light is placed on a finger and a photo-detector placed right across the source detects the transmitted light reflected back. Shortly after the systole, the amount of blood in the arteries increase, thus reflecting it on the intensity of received light which increases too. The contrary occurs during the diastole, where the amount of blood in the arteries decreases leading to a decrease in the light observed by the photo-detector. Blood flowing characteristics are unique identifiers specific to different persons while they are similar enough to recognize the same person [2, 3, 4], keeping a strong relationship with person's anatomy and physiology as with the heart size and its dynamics.

Most of the approaches in the literature for biometric pulse identification rely both on involving Electrocardiography (ECG), based on the electrical activity of the heart, and on a carefully design, segmentation and extraction of expert features from the pulse signal [5, 6]. A decoupled approach which comprises mainly two stages is usually described [2, 7, 8]. Firstly, biomarkers or features are extracted from the pulse ECG or PPG signals, also known as front-end processing. Then, template feature vectors feed a second stage that performs model learning. Nonetheless, such features are designed by hand and strongly depend on a high expertise both on the knowledge of the addressed task and on acquisition nature of the pulse signal itself. For instance, in [2] an experiment on a group of 17 subjects was performed, where the authors studied four time domain characteristics, as time intervals, peaks and slopes from the PPG signals reporting successful accuracy rates of 94% for human verification. In the work of [7], feature extraction on the PPG, ECG, EEG signals was performed based on eigenvector methods. Spachos et al. [3] studied four feature parameters, peak number, time interval, upward slope and downward slope. The study from [4] is intended for exploring the time domain features acquired from its first and second derivatives, where a group of 40 features were extracted and ranked based on a k-nearest neighbor algorithm. The authors in [8] perform a comparison of three methods based and proposed the pulsatile beat-by-beat correlation analysis, the rejection or acceptance of subject is performed based on the maximum similarity. Finally, more recent works [9] make use of Deep Belief Networks and Restricted Boltzman Machines as classifiers. With the advent of Deep Neural Network architectures, such as convolutional based neurons, end-to-end processing pipelines are gaining popularity by building architectures capable of learning features directly from raw data. For instance, in computer vision [10] or speech processing [11, 12] novel feature learning techniques are applied directly on the raw representations of images and audio, avoiding the signal parameterization or any other prior preprocessing.

This paper presents a new human verification approach using photoplethysmogram (PPG) signals and deep neural network modelling. The novelty of this work resides on the use of an end-to-end deep neural network architecture for both automatic extraction of biomarkers and low complexity allowing high continuous authentication rates. The proposed front-end, based on a Convolutional Neural Network, is jointly trained together with a dense neural net. Such architecture allows for a joint optimization of the extracted patterns

ISBN 978-90-827970-1-5 © EURASIP 2018

while maximizes the verification of the subject's identity. In addition and for development and evaluation purposes a new database, named as *PulseID*, is collected within a regular office environment, comprising 43 subject's IDs and their PPGs signals. Our proposed verification approach through neural network learning and classification appears to be viable as reported by the experiments performed on the Troika [13] and *PulseID* datasets. The results are encouraging, reaching AUCs around 83.1% by trials lasting just 1 second, both showing the potential of learned PPG biomarkers as a stand alone biometry and allowing for a continuous authentication in real-world scenarios.

2. PPG BASED VERIFICATION METHODOLOGY

2.1. Datasets

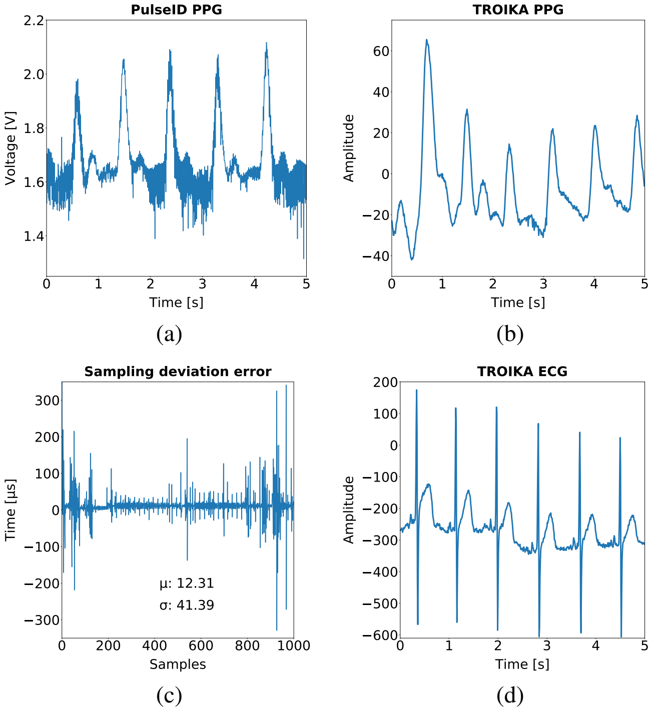
Two different datasets are employed in this work to conduct person verification experiments through PPG signals. Firstly, a new corpus was collected aiming to fulfil a need of, to the best of authors' knowledge, a public domain dataset specifically created for PPG biometric identification. For such a purpose, the authors collected a new PPG dataset, named as *PulseID*[1], in a quiet office environment. Secondly, aiming to verify the robustness of our approach in more challenging conditions, the Troika [13] dataset is used. In contrast, Troika recordings are acquired for subjects on a treadmill, walking and running at different speeds.

For the PulseID data acquisition, the pulse sensor described in [14] is employed. It is essentially a photoplethysmograph, a well known medical device used for non-invasive heart rate monitoring, consisting of a green LED and a photo-detector. The heart pulse signal that comes out of the pulse sensor is an analog fluctuation in voltage, with associated waveform known as photoplethysmogram or PPG, see figures 1(a) and (b). The pulse sensor responds to relative changes in illuminance. For a sensor placed in the subject's skin, the reflected light back to the photo-detector changes during each pulse due to blood flowing what is perceived as variations in the voltage signal. A Raspberry Pi 3 board was employed for hardware acquisition together with a popular analog to digital converter (ADC) MCP3008, accounting for 10 bits. The process of data acquisition was provided by 43 volunteers (31 male and 12 female) with ages ranging from 22 to 55. Subjects were seated down in a calmed, relaxed and quiet office environment while the recordings. The PPG sensor was attached to the fingertip of the right index finger by a belt. PPG acquisitions, lasting roughly one minute, were recorded from each subject and repeated 5 times along the same session. PPG analog signal was sampled at 200 Hz rate. For doing so, a python code was developed aiming to perform sampling synchronization while reading from the ADC and ensuring a tolerant averaged sampling rate deviation of $\mu = 13.32$ μs and $\sigma = 202.58$ μs per subject, see plot (c) in the figure 1 for an example. For comparison purposes, an ECG waveform from Troika is depicted in fig. 1(d). It is worth noting that no pre-processing is performed to the raw PPG acquired signals. It can be seen in the higher noise levels present in the acquired PPGs, where various artifacts are expected to be found: like analog circuit noises or medium illuminance changes, respiration or base deviation arising from movement.

[1] The *PulseID* dataset is available upon request from the authors and agreement of EULA for research purposes.

Table 1: Summary stats for both databases, Troika and PulseID. The "duration" column stands for the average duration in seconds of the total acquired samples per subject

Dataset	Subjects	Gender (m/f)	Duration
PulseID	43	31/12	240s.
Troika	20	20/—	317s.



(a)

(b)

(c)

(d)

Fig. 1: Five seconds PPG excerpts from PulseID (a) and Troika (b) databases. (c) Time sampling deviation in μ seconds from the same PPG signal in (a). In (d), a five seconds ECG excerpt from Troika

In addition to PulseID database, the Troika dataset is employed to verify the robustness of our approach. Biometric identification using PPG should be possible even when the subject is in heavy physical motion. Therefore, Troika introduces a suitable database to benchmark learning models in practical day to day situations, by presenting higher heart signal variability and physical motion artifacts, is in theory a more challenging scenario compared to a relatively quiet office environment. During Troika recordings, subjects walked or ran on a treadmill at different speeds. The data was collected from 20 male subjects with ages ranging from 18 to 35. For each subject, the PPG signals were recorded from wrist by two pulse oximeters but only the first PPG channel is used in this work. The pulse signals were sampled at 125 Hz, see [13] for further details.

2.2. End-to-end biomarker learning

Convolutional Neural Networks (CNN) have become broadly applied reporting great success for instance in image recognition tasks [15, 16]. In the same sense, our deep CNN-based feature learning architecture makes use of local filtering and feature pooling, used at the output of the convolutional layers. The CNN architecture that we used as a basis for all our experiments is depicted in figure 2.

The end-to-end is validated and tested using 31 target and 12 impostor subjects with total of 43 subjects for the *PulseID* database. For the case of Troika, 15 target and 5 impostors accounting for a total of 20 subjects were used. The total number of enrolled subjects is 35 and 15, respectively. In the training phase, the waveform is homogeneously segmented in chunks of duration 1 second for each

ISBN 978-90-827970-1-5 © EURASIP 2018

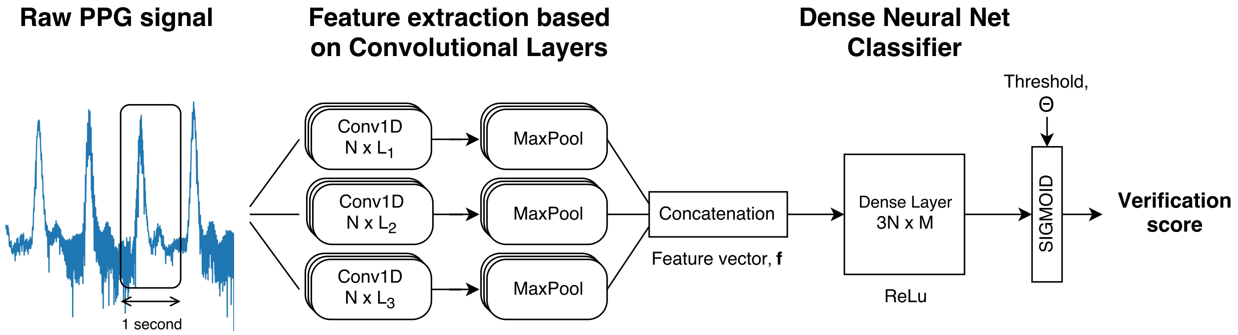


Fig. 2: Proposed Convolutional Neural Network architecture for end-to-end user verification using raw PPG signals. First, the raw signal is filtered by three parallel 1-D convolutional layers composed of N filters of lengths $L_{1,2,3}$ followed by a global max-pooling operation. The resulting $3N$ features are then concatenated into the feature vector $\mathbf{f}$, which is used to perform the classification using a dense layer of dimensions $3NxM$ and the final layer of 1 output. ReLU activation function is used across all layers but the output layer, where a sigmoid activation is used to predict the verification score.

of the subjects. The table 2 reports on the partition set in terms of target and impostor trials for the case of excerpts lasting 1 second. Note that in a 60 bpm (beats per minute) Heart Rate there is one beat each second, there is no data-partition preprocess to ensure that a full peak is taken because every chunk contains a different signal, unless the subject Heart Rate is stable at 60bmp during all the acquisition. A combination of pairs (target,impostor) from the *Train* and *Validation* sets are used for network training and validation, performing parameter updating based on cross-entropy loss computed on the *Validation* set. The *Develop* set is used for final network testing and threshold selection but note that impostor trials are drawn from the same pool of identities than from previous sets. Finally, the *Test* set composed of unseen impostor subjects is employed for the fair assessment of the end-to-end proposed approach. Such data partitioning aims to prevent biasing and resembles a real use-case scenario in which cross-validated model is benchmarked against new enrolled users.

Note that the CNN-maxpooling feature learning architecture applies 1-dimensional convolutions and pooling operations performed along the time axis as previous works in [11]. Let's assume the PPG signal input to the CNN is a vector, $\boldsymbol{x}$, whose elements are raw PPG samples $\boldsymbol{x} = [x_k, x_{k+1} \ldots x_{k+K}]$ where x_k is the PPG sample shifted by a stride. In this work we used a value of 1 for time shifting and the x_k sample with a fixed size from 1 second at 200 samples per second. The activations at the first convolutional layer comprise $N = 6$ filters and we denoted them as $\boldsymbol{h}_n = [h_1 \quad h_2 \cdots h_N]$. Therefore, the convolutional layer operation can be seen as a convolutional operation of each filter on the input raw PPG,

$$h_n = \theta\left(\boldsymbol{w}_n \boldsymbol{x}^T + b_n\right),$$

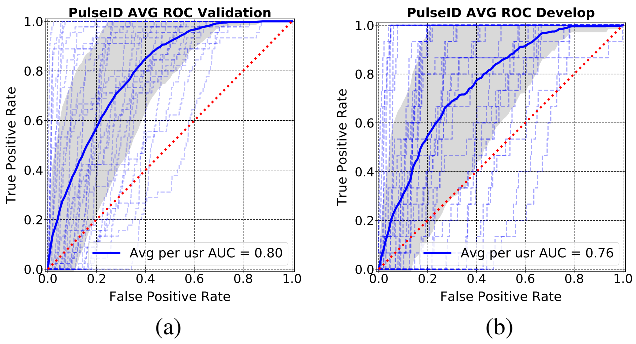
where $\theta(x)$ is the activation function corresponding to Rectified Linear Units (ReLU), $\boldsymbol{w}$ is the weighting vector and b_n the bias term for filter h_n. Following the convolutional filters, max-pooling layers perform local temporal max operations over the input sequence, selecting the maximum in a window of d size. More formally, the transformation at starting sample vector n, $\boldsymbol{c}_k^n$, corresponding to the filter output sequence of the first convolutional layer and jth filter is:

$$\max_{k-\frac{(d-1)}{2}\leq s\leq k+\frac{(d-1)}{2}} c_s^n$$

The pooling operation compacts even more the original signal by computing some stats, commonly such as maximum, mean and variance, from the CNN output. For this work maximum pooling is used by selection of the maximum values from the CNN filter outputs. Next, a flattering operation is performed, see figure 2, that aims at stacking together all the CNNs outputs, creating a feature vector ready to be presented to the network classifier. In overall, the end-to-end architecture comprises a total of 4 layers. In the input, a convolutional layer with different amounts of filters and lengths (see figure 2) followed by a max pooling layer. At the back-end, a fully connected neural net composed of 2 layers with 256 units.

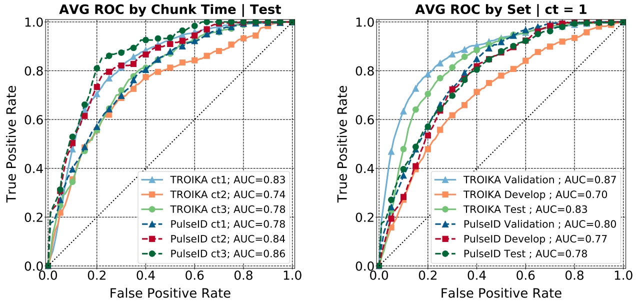
Table 2: Partition data for the different sets. Trials are expressed in seconds of signal and averaged per subject Since the trial size of the experiments showed is one second, the number of *Target* and *Impostor* data corresponds to number of trials or seconds

Dataset	Label	Train	Validation	Develop	Test
PulseID	Target	135	45	30	30
	Impostor	5,220	1,740	1,890	2,880
	#Subjects		31		12
Troika	Target	144	80	48	48
	Impostor	2,014	1,119	1,343	1,545
	#Subjects		15		5



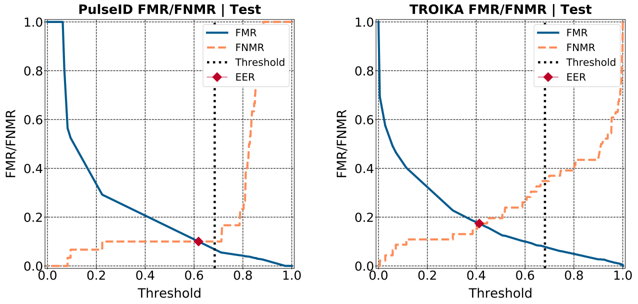
(a) (b)

Fig. 3: Average ROC for validation (a) and develop (b) sets using $N = 6$ filters for each filter sizes of $L_{1,2,3} = 50, 30, 20$. The painted area corresponds to the area within the standard deviation of the AUC. Dashed lines stands for each subject AUC curves



(a) (b)

Fig. 4: Average ROC comparison according to (a) the size of the chunk time, ct=1, 2, 3 s. and (b) set of data. The curves have been computed as in figure 3, averaged for all the subjects



(a) (b)

Fig. 5: False Match Rate (FMR) and False Non-Match Rate (FNMR) ratio plots as function of threshold Θ. Threshold line Θ, see fig. 2, corresponds to the operating point where the FMR is below 0.1 and the FNMR is minimum

ReLu units are employed in all layers, including CNNs, except for the output layer of the dense network, where we used a sigmoidal unit. The dense layer is employed as a back-end for the modeling of the salient features computed by previous convolutional steps and a Sigmoid activaction function is used in the output layer. It is worth to note that no dropout is used during network training.

The framework [17] has been developed in Keras [18] and using Tensorflow [19] as back-end. We do not perform an exhaustive search of network parameters and we restrict experiments by using few learned biomarkers. For instance, we compute 15 features before the dense layer for the reported 1s. experiments, see vector **f** in fig. 2. The network is trained using Stochastic Gradient Descent (SGD) attending to binary cross-entropy as a loss function and accuracy as a metric, with mini batches of size 270 composed of 135 target trials and 135 impostor ones. Given a *Train* set of PPG excerpts from a subject, at each training mini batch, the impostor samples are randomly picked up from the available pool of impostor chunks. Thus, in each training iteration, new impostor data is seen as an intent to maximize variability, see table 2. An early stopping criteria is also defined in order to speed up the training, yielding in most of the cases to few tens of training mini batches before reaching *patience* steps. The figure 3 shows the ROC curves, per *Validation* and *Develop* sets in PulseID and averaged per subjects, solid line, and its standard deviation, shadowed area. For the sake of comparison, the same curves are depicted in figure 4(b) per each dataset and partition.

3. EXPERIMENTAL RESULTS AND DISCUSSION

Although an exhaustive search of the best network architectures or a fully tuning of parameters is not performed, we experiment with different window and filter sizes. For the reported figures, we select 1, 2, 3 second excerpts extracted from original raw PPG, homogeneously segmented and with no overlap for testing trials. The experiments are performed in PulseID data and best values, in terms of number of filters and size, are directly applied in Troika. Homogeneous segmentation of the input PPG likely degrades system performance due to few samples are taken into account for training, see table 2. However, it could be easily bypassed, e.g., by a randomly picking of excerpts thus increasing samples and segmentation variability in train and test. The figure 4(a) and table 3 report on the system performance for different trial sizes, ranging from 1 to 3 seconds. We can observe the generalization of validation results both in *Develop* and *Test* sets, showing high AUC values even for 1s. trial condition, AUC=0.78 and 0.83 per each dataset. Note the higher AUC degradation in Troika compared with the PPG data captured in the office condition and the AUC trend observed by increasing the *ct* time, not observed in Troika likely due to motion artifacts. In overall, the results support the suitability of the end-to-end architecture in both datasets, although as observed in ROC curves fig.3, some subject's AUC present a not satisfactory behaviour suggesting more experimentation to understand possibles sources of such variability.

Table 3: Average AUCs for all subjects within the same experiment: $N = 6$ filters for each filter sizes of $L_{1,2,3} = 50, 30, 20$. the ±variation corresponds to the AUC's standard deviation

Dataset	Trial size	Validation	Develop	Test
PulseID	1s.	0.80±0.16	0.77±0.19	0.78±0.20
	2s.	0.81±0.16	0.76±0.22	0.84±0.19
	3s.	0.84±0.15	0.78±0.20	**0.86±0.17**
Troika	1s.	0.87±0.09	0.70±0.16	**0.83±0.12**
	2s.	0.73±0.30	0.66±0.21	0.74±0.24
	3s.	0.85±0.14	0.71±0.16	0.78±0.18

Another parameter to determine in authentication systems is the operating point or decision threshold. It controls the trade-off between usability, minimum FNMR, and security, minimum FMR. The fig. 5 reports on the Equal Error Rate (EER) 0.1 (a) and 0.174 (b) for a specific user in both datasets. Related to the algorithm complexity for trial decision, taking into account L_i size of filters, pooling and dense layers operations, and 1s. trial lead to a number of MAC operations [20] around $26K$, that translates into roughly 20ms for a Raspberry in order to perform person authentication every second.

4. CONCLUSIONS

An end-to-end architecture based on CNN is proposed to offer biometric authentication using learned biomarker directly from PPG raw signals. We reported evaluation results of the performance of our approach in two different datasets, Troika and PulseID. Our end-to-end authentication approach and automatic learned biomarkers show a remarkable potential as authentication biometric method. Trial size dependent experiments, reported AUCs ranging [78.2%, 86.4%] and [73.8%, 83.2%], averaged among target subjects on PulseID and Troika datasets, respectively. Furthermore, the proposed system results in a low complexity that permits for continuous authentication in real-world scenarios.

5. REFERENCES

[1] A. V. Challoner and C. A. Ramsay, "A photoelectric plethysmograph for the measurement of cutaneous blood flow," *Phys Med Biol.*, vol. 19(3), pp. 317–28, 1974 May.

[2] Y. Y. Gu, Y. Zhang, and Y. T. Zhang, "A novel biometric approach in human verification by photoplethysmographic signals," in *4th International IEEE EMBS Special Topic Conference on Information Technology Applications in Biomedicine, 2003.*, April 2003, pp. 13–14.

[3] P. Spachos, J. Gao, and D. Hatzinakos, "Feasibility study of photoplethysmographic signals for biometric identification," in *2011 17th International Conference on Digital Signal Processing (DSP)*, July 2011, pp. 1–5.

[4] A. R. Kavsaoğlu, K. Polat, and M. R. Bozkurt, "A novel feature ranking algorithm for biometric recognition with ppg signals," *Computers in Biology and Medicine*, vol. 49, no. Supplement C, pp. 1 – 14, 2014.

[5] S. A. Israel et al., "Ecg to identify individuals," *Pattern Recognition*, vol. 38, no. 1, pp. 133 – 142, 2005.

[6] H. P. da Silva, A. Fred, A. Lourenço, and A. K. Jain, "Finger ecg signal for user authentication: Usability and performance," in *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, Sept 2013, pp. 1–8.

[7] Elif Derya Übeyli, Dean Cvetkovic, and Irena Cosic, "Analysis of human ppg, ecg and eeg signals by eigenvector methods," *Digital Signal Processing*, vol. 20, no. 3, pp. 956 – 963, 2010.

[8] T. Choudhary and M. S. Manikandan, "Robust photoplethysmographic (ppg) based biometric authentication for wireless body area networks and m-health applications," in *2016 Twenty Second National Conference on Communication (NCC)*, March 2016, pp. 1–6.

[9] V. Jindal, J. Birjandtalab, M. B. Pouyan, and M. Nourani, "An adaptive deep learning approach for ppg-based identification," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug 2016, pp. 6401–6404.

[10] Q. V. Le, "Building high-level features using large scale unsupervised learning," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, May 2013, pp. 8595–8598.

[11] C. Segura et al., "Automatic speech feature learning for continuous prediction of customer satisfaction in contact center phone calls," in *Advances in Speech and Language Technologies for Iberian Languages*, Cham, 2016, pp. 255–265, Springer International Publishing.

[12] Y. Gong and C. Poellabauer, "How do deep convolutional neural networks learn from raw audio waveforms?," 2018.

[13] Z. Zhang, Z. Pi, and B. Liu, "Troika: A general framework for heart rate monitoring using wrist-type photoplethysmographic signals during intensive physical exercise," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 2, pp. 522–531, Feb 2015.

[14] J. Murphy and Y. Gitman, " PulseSensor Open Hardware ," http://pulsesensor.com/, 2017, [Online; accessed 19-October-2017].

[15] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, 1995.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[17] G. Cortès, J. Luque, J. Fabregat, and J. Esteban, " PulseID Keras code for development and testing purposes," https://guillemcortes@bitbucket.org/guillemcortes/pulseid-eusipco, 2018.

[18] F. Chollet et al., "Keras," https://github.com/fchollet/keras, 2015.

[19] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, Software available from tensorflow.org.

[20] V. Sze, Y. Chen, T. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

ISBN 978-90-827970-1-5 © EURASIP 2018