Master thesis on Intelligent Interactive Systems

Universitat Pompeu Fabra

Correlation of speech/non-speech events with photo-plethysmographic (PPG) signal

Guillermo Cámbara Ruiz

Supervisor: Jordi Luque

Co-Supervisor: Mireia Farrús

June 2019

Master thesis on Intelligent Interactive Systems

Universitat Pompeu Fabra

Correlation of speech/non-speech events with photo-plethysmographic (PPG) signal

Guillermo Cámbara Ruiz

Supervisor: Jordi Luque

Co-Supervisor: Mireia Farrús

June 2019

Contents

Dedication

I would like to dedicate this work to Maria, whose unconditional love pushes persons towards their better selves.

Acknowledgement

I would like to express my sincere gratitude to:

- Jordi Luque
- Mireia Farrús
- My family

Abstract

The use of photoplethysmogram signal (PPG) for heart monitoring is commonly found nowadays in smartphones and wrist wearables. Besides heart rate or sleep monitoring common usage, it has been proved that information from PPG can be extracted for other uses, like person verification, for example. In this work, we evaluate whether if speech/non-speech events can be inferred from fluctuations they might cause in the pulse signal. In order to do so, an exploration on end-to-end convolutional neural network architectures is done for performing both feature extraction and classification of the mentioned events. The results are motivating, detecting speech in PPG signal with a 68.2% AUC using the best performing architecture. On the other hand, a first experiment on speaker's voice pitch detection is done, in order to check if a prosody marker such as pitch variation could be present in PPGs, but such clue is not clearly found in the results obtained. Nevertheless, the correlation between speech and PPG signal is proven and the way is paved for further experiments on this topic.

Keywords: Photoplethysmogram signal; PPG; Speech detection; Prosody markers; Convolutional neural networks

Chapter 1

Introduction

The advent of Artificial Intelligence (AI) and concretely deep learning has allowed the design of highly accurate classification systems, which require a lesser effort on the feature extraction process from data. Because of this, such methods are widely used nowadays in the speech processing field [1], for tasks such as automatic speech recognition or speech synthesis, just to name a few of them. However, the capability of deep learning to mine relevant information from large data sets is so big, that novel applications are being designed based in it. In the case of this work, an exploratory study with Neural Networks is done in order to find out if it is possible to detect speech just from heart beat signal, or even finding pitch information, which would be a first step towards finding prosodic cues in future experiments.

1.1 Motivation

Biometric sensors are embedded in many electronic devices nowadays, like smartphones or smartwatches, just to name a few of them. Typically these sensors have been used to retrieve information like heart rate, blood oxygen level or fingerprint identification, for healthcare and security applications.

However, recent studies have found that biometric signals obtained from such applications can be processed by neural networks, in order to extract further information

and enhance the common use case possibilities for biometrics in wearables.

For instance, it has been shown that person authentication can be performed just with photoplethysmography (PPG) signal, feeding it to an end-to-end Convolutional Neural Network architecture, which is able to automatically extract relevant features in the signal and identify from which person it is coming from [2].

Such findings motivate this work, where it is intended to develop a deep learning-based architecture that is able to extract relevant information from PPG signal as well, in order to open up for further application possibilities. Particularly, the aim is to find out the correlation of speech/non-speech events with PPG signal. Being able to obtain such information would allow to develop cost-friendly applications, that would use PPG signal for tasks like speech detection, ASR enhancement or even word recognition.

1.2 Objectives

The main objective of this work is to perform an exploratory study on the correlation of speech/non-speech events with heart beat PPG signal, using deep learning architectures. In other words, the following question is addressed: is it possible to detect if a person is speaking or not just by feeding its heart beat PPG signal to a Neural Network? If so, which could be the nature of speech representation in PPG signals? Would there be any chance to find prosody markers in further experiments? These questions attend directly the first objective of this work, which is to develop a Neural Network architecture capable of classifying a PPG signal sample as speech or non-speech, depending on if a speaker was speaking or not when that sample was taken. The hypothesis is that speech could be found as a fluctuation in the PPG signal, caused by an additional oxygen consumption during speech production [3], or just as some form of noise due to acoustic vibrations captured by the sensor [4, 5]. However, if this first correlation is achieved, then it would be proven that PPG signal could carry speech information, and next steps could be taken, like trying to find prosodic traits in it.

Because of the complexity of the task of finding prosodic traits in PPG signal, the pitch of the voice is defined as the first variable to experiment with, since its variation during speech is a characteristic of prosody. In acoustic terms, the pitch of the voice corresponds closely to the fundamental frequency, which is typically different for men (between 85 and 155 Hz) and women (from 165 to 255 Hz). If acoustic vibrations from speech are captured by the PPG sensor, then this fundamental frequency might be represented in the PPG signal. Being so, since the sampling rate of the sensor is 200 Hz, it would be expected to capture the fundamental frequency of some male subjects speaking under 100 Hz, because of Nyquist's theorem. Therefore, this fundamental frequency would aid the Neural Network to perform gender classification with PPG signals as input. So on, three gender classification experiments would be done, one with speech only PPG samples, another one with non-speech only PPG samples and the third one with both of them. It would be expected that the classification accuracy of the experiment with only speech samples should be higher than the one with non-speech samples. This is because these samples would have additional information from the fundamental frequencies of men's voices. Besides, gender classification with PPG signal and deep learning is also a novel experiment in its own. The authors in [6] were able to classify gender with PPGs, but using traditional feature extraction and k-nearest neighbours algorithm.

If these objectives are fulfilled, the potential use of PPG signal for tasks like speech detection, or ASR enhancement would be proved, also opening up for more types of experiments regarding speech events and prosody traits in PPG. However, it must also raise a flag about data privacy, since sensitive information from a smartphone/wearable user could be extracted from this biometric data.

1.3 Structure of the Report

This report contains four main chapters, besides the Introduction itself: State of the art, Methods, Results and Discussion.

To begin with, the State of the art section presents the state of the latest works

involved in the topics for this thesis, such as speaker identification with PPG signal, speech detection with unconventional devices or noise detection in ECG signals, for example.

Within the Methods section, there is a deeper explanation of the dataset and the architectures used in the work, as well as details for more theoretical aspects of the hypotheses here presented, involving topics like PPG, deep learning or prosody, for example. Furthermore, the experimental setup is explained as well, for every experiment designed to fulfill the objectives stated in the previous section.

In the Results section, the obtained results are presented, with the aid of tables and figures. Such findings are presented with a brief interpretation, which shall be extended in the Discussion section.

To conclude with, the Discussion section extends the explanation of this work's results, as explained above. Besides, it presents the final conclusions, linking the relevant results for all the proposed experiments.

Chapter 2

State of the art

Up to the author's knowledge, there has not been any previous work trying to find the correlation between PPG signal and speech, with or without deep learning methods. Being so, it is needed to introduce the state-of-the-art on the topics converging in this work, such as speaker identification with PPG signal, speech detection with unconventional devices, oxygen consumption on speech production or Deep Neural Networks applied to noise detection in ECGs. The purpose of gathering the state-of-the-art for these topics is to find out how a combination of such methods could help design a system able to recognize speech from PPG signal.

To begin with, the results found by Luque et. al. in [2] showed that PPG signal contains information about a speaker's identity that can be extracted with a CNN architecture, without need of expert knowledge to extract relevant features from such signal, presenting an end-to-end system. Previous works that found correlations between PPG signal and person verification needed an additional effort to extract these features. For example, in works like [7] or [8] the authors studied time domain characteristics like time intervals, peaks, upward and downward slopes in PPG signal.

Being so, these results suggest that PPG signal could carry also information about the speech itself, and that using a CNN-based architecture might ease the feature extraction effort. Thus, a system that would eavesdrop PPG signal could be de-

signed, being able to tell if a person is speaking or not because of variations in it caused by difference in oxygen consumption, acoustic vibrations, etc. Up to the author's knowledge, there has not been any previous work trying to find the correlation between PPG signal and speech with such deep learning methods. However, there have been works related to speech recognition with unconventional devices (other than microphones), like gyroscopes [4] or accelerometers [5].

Michalevsky et. al. [4] found that gyroscopes capture acoustic vibrations from speech under frequencies of 200 Hz, which can be used to detect speech, identify speakers, or even parse such speech. In order to prove this, they used Short Time Fourier Transform (STFT) and Mel-Frequency Cepstral Coefficients (MFCC) as features, and Machine Learning classification methods like Support Vector Machine (SVM), Gaussian Mixture Model (GMM) and Dynamic Time Warping (DTW).

In a similar manner, Matic et. al. [5] found that acoustic vibrations leaked into an accelerometer signal as well, allowing for speech activity detection. Being so, they performed feature extraction using the Fast Fourier Transform (FFT), and used various classification algorithms, like SVM or Naïve Bayes, for example, which yielded successful results.

Thus, such results show that low frequency information from speech can be captured by detectors other than microphones, which leveraged by proper feature extraction and Machine Learning algorithms allow some speech processing tasks. Therefore, this low frequency information could be obtained from a PPG signal as well, if the sensor is sensitive enough. However, besides acoustic vibrations, an interesting indicator of speech might be the difference in oxygen consumption between speech and rest, measured by the PPG signal. Theoretically, speaking should consume some extra oxygen, because of an additional biomechanical effort and an interruption of the resting breathing flow. This should be represented as some form of fluctuation or noise in the PPG signal.

Moon and Lindblom performed two experiments in order to shed light on how oxygen consumption is affected by speech production [3]. In the first experiment, they

proved that oxygen consumption increases when the vocal effort (how loud a person speaks) is higher. On the other hand, they also showed that a higher frequency of pronounced syllables yields a higher oxygen consumption as well. The first result reinforces the hypothesis that speech or non-speech events could be classified from a PPG signal because of the oxygen concentration in blood. Moreover, the second experiment related to the syllables frequency gives a clue that some prosody traits might be obtained as well.

All in all, the mentioned state-of-the-art suggests that the act of speech causes some fluctuations because of acoustic vibrations and/or oxygen consumption, that might be captured by a detector such as a PPG sensor. It is interesting to find out which deep learning architectures can be used for detecting noise in a pulse signal. In [9] several CNN-based architectures are tried, in the task of labeling how noisy are some frames extracted from an electrocardiogram (ECG) signal. The best architecture found by the authors is a 16-layer CNN adapted from the VGG16 network [10].

To sum up, the starting point for this work is to try to detect such noise or fluctuations in PPG signals, making use of the proposed end-to-end CNN for speaker identification and the VGG16-like network for noise detection in ECGs. A proper adaptation of these architectures should fit the proposed problem of detecting speech in PPG signal. Find here below a further explanation of the mentioned state-of-the-art works.

2.1 Speaker Identification with PPG signal

Due to the increasing popularity of wearable sensors, user identification through heart signal monitoring has raised the interest in the research community. Typically, research in this field has involved the usage of Electrocardiography (ECGs) and careful extraction of relevant biomarker features from it. However, the work presented in [2] suggests the usage of deep learning method, concretely Convolutional Neural Networks (CNNs), for automatic extraction of these biomarkers in PPG signal. Thus, an end-to-end architecture is proposed, which is able to perform

user identification with a 78.2% AUC for the *PulseID* dataset, taking raw PPG signal as an input. Find the exact architecture in figure 1, extracted from the reference [2].

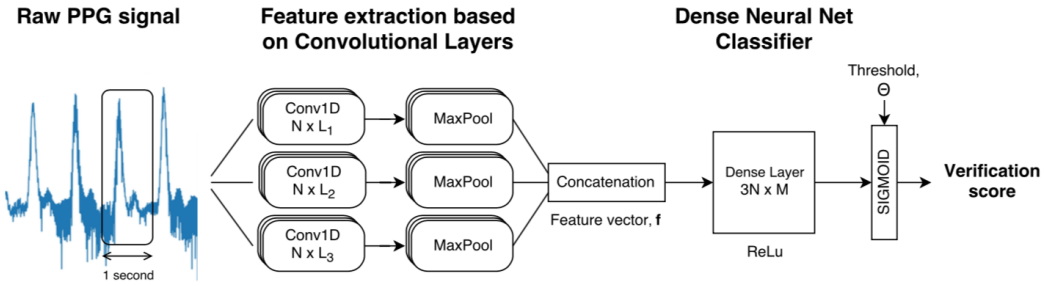

Figure 1: CNN Architecture used for end-to-end user identification in [2].

The raw PPG signal is processed by three parallel convolutional layers (Conv1D), with each one of them having $N = 6$ filters of lengths L_1, L_2, $L_3 = 50, 30, 20$. The output vectors from each layer are filtered by a global max-pooling operation, and finally concatenated into a feature vector of length 3N. This feature vector is passed into a dense neural net classifier of dimensions $3NxM$ (2 layers of 256 units), which finally plugs the result into a sigmoid activation that predicts the verification score. ReLU activation functions are used for all layers except the last one using the sigmoid.

The proposed architecture is trained with data from 31 different subjects, split in *training*, *validation* and *test* sets. The first two are used for training the parameters in the network, performing parameter updating based on cross-entropy loss.

2.2 Indirect Speech Detection

The work presented in [4] involves speech detection and parsing with a MEMS gyroscope. Gyroscopes are sensitive enough to measure acoustic vibrations, even though the resulting signals contain only low frequency information (< 200 Hz). It is shown that it is possible to detect speech and even parse such speech, using signal processing and machine learning techniques.

However, because of the limited sampling rate, it is not possible to fully reconstruct

a comprehensible speech from the gyroscope signal. Even though, from a set of 10 speakers, there is a 50% success rate in speaker identification. On the other hand, parsing of a correct digit between 0 and 9 is done with a 65% accuracy for speaker dependent case and up to 26% recognition rate for speaker independent case. Furthermore, combining the signals from two gyroscopes, the accuracy for speaker dependent digit recognition task increases up to a 77%. These results are achieved by extracting the Mel-Frequency Cepstral Coefficients (MFCC), which employ the Cepstrum transformation, separating the signal originated by air passing through the vocal tract from the effect of the vocal tract. Furthermore, the Mel-scale compensates for the non-linear frequency response of the human ear. On the other hand, the Short Time Fourier Transform is computed as well, which basically windows the signal in short overlapping pieces and computes the FFT over them, obtaining a spectrogram of the signal.

Once these features from MFCC and STFT are extracted, three different classifiers are used in this work. Support Vector Machine (SVM) is used to distinguish male and female speakers, and also to distinguish between multiple speakers and to recognize words from a limited dictionary. Furthermore, a different Gaussian Mixture Model (GMM) is trained for each group during the training stage. During testing, a match score for each group is obtained from every sample, and the sample can be classified as belonging to a certain group regarding the maximum score. Finally, Dynamic Time Warping (DTW) is used to match time-dependent features in presence of misalignment, which is common in word recognition tasks, since the samples usually differ in length, resulting in different number of segments for feature extraction.

For speaker identification, every gyroscope recording is transformed to WAV format, up-sampled to 8KHz, and a silence removal algorithm is employed for cleaning the signal of unvoiced segments. Statistical features from the first 13 MFCC computed on 40 sub-bands are used, and the STFT features are computed with a window of 512 samples, which corresponds to 64 ms. Using STFT with DTW algorithm yields the best results, obtaining a 84% accuracy for gender identification. Also, for speaker

identification, 50%, 45% and 65% accuracies are obtained for mixed female/male, female and male speakers respectively.

On the other hand, for the task of digits recognition, **TIDIGITS** corpus was used. This corpus contains 220 recordings, each one being an isolated digit (from 0 to 9, including the syllable "oh"), said by 10 different speakers, 5 males and 5 females. For this task, the best results were given again using STFT features with DTW classifier, with 17%, 26% and 23% success rates for mixed female/male, female and male speakers respectively.

Furthermore, the authors in [5] propose an accelerometer-based speech detection method, which detects phonation-caused vibrations at the chest level, targeting frequency range approximately between 100 and 200 Hz. 21 subjects are asked to read out loud some articles from newspapers during 2 minutes, while having an accelerometer attached to the chest with an elastic band. On the other hand, recordings are done as well while the subjects do mild physical activities without speaking.

The frequency spectrum is obtained from the signals, applying the Fast Fourier Transform (FFT) to compute the Discrete Fourier Transform (DFT), and finally the power spectral density. For each 10-seconds signal frame, the power spectral density is computed from the sum of spectral densities for each 2 seconds. Besides, some specific parameters are used for characterizing the spectral density, such as mean, maximal, minimal and integral values from different frequency ranges.

Regarding classification algorithms, SVM, Naïve Bayes, and Naïve Bayes with kernel density estimation and k-NN are tried. The last one applied on integral and mean values of the components between 80 Hz and 256 Hz yields the highest classification accuracy. The system is able to detect voice with a 93% accuracy, in the case where speakers were speaking. On the other hand, the algorithm only wrongly classified signals from mild activities as speech the 19% of times.

2.3 Oxygen Consumption during Speech and Noise Detection in ECGs

The experiments designed in [3] serve the purpose of measuring oxygen consumption with two variables: the amount of vocal effort (loudness) and the syllables rate (syllables per second).

For the first experiment, the subjects are asked to count to eight, in synchronization with a metronome beat. They repeat this procedure three times, one at a different vocal effort level: *soft*, *normal* and *loud*. Soft and normal results are very similar, but the oxygen consumption for speaking loud is significantly higher.

On the other hand, during the second experiment the subjects pronounce the syllabel "[sa]" in synchrony with the metronome beat, at a constant vocal effort, with two different speaking rates: *normal* and *fast*. Fast rate means that the syllable frequency is duplicated. Under this late condition, the subjects use more oxygen, consuming around 25% more than in normal pace.

Such variations in oxygen consumption could be found in PPG signals as some form of apparent noise, so it is interesting to study how other authors have dealt with noise in PPG signals, or at least with ECGs, since they are similar.

For the task of detecting noise in ECGs signals, carried out by [9], several CNN architectures are tried. At first, signal windows of 2.6 seconds are processed, using CNN models with few layers, the first one having two one-dimensional convolutional layers followed by two fully-connected layers. Suspecting that more layers are needed for this task, the authors add two more convolutional layers.

However, it is seen that more context than 2.6 seconds would be needed to classify a sample as noisy or not, so 10 second samples are used, with a deeper architecture. This time, a VGG-like architecture is used, but inverting the order of filter numbers. Being so, instead of increasing the number of filters for every subsequent layer, they are decreased instead, like can be seen in figure 2. With this architecture, the authors

achieve the best noise classification result, with a 97.7% AUC.

It is worth mentioning that VGG-like architectures [10] are widely used in several tasks for different fields, such as image or sound classification, for example. These are very deep Convolutional Neural Networks, with increasing number of filters at every layer and small kernel sizes.

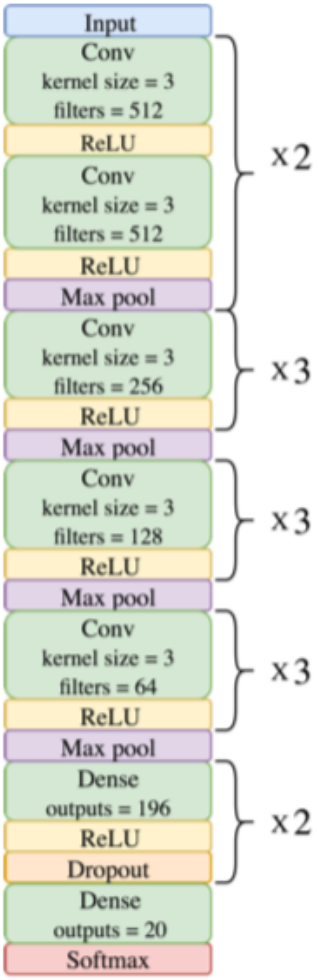

Figure 2: Inverted VGG Architecture used for noise detection in ECG in [9].

Chapter 3

Methods

The methodology for this works is introduced in this chapter. A further explanation on the chosen data set can be found here, as well as a description on the used architectures and how they are trained. Besides, the procedures for each experiment are also presented here.

3.1 Dataset

For the purpose of finding the correlation between PPG signal and speech events, a new dataset has been collected, named as *PulseID*, in a quiet office environment.

The sensor described in [11] is used for capturing the PPG signal. This sensor is a photoplethysmograph, which consists of a green LED and a photo-detector. It is placed in the subject's finger, and the reflected light causes a fluctuation in voltage, which is correlated to the variation of red light caused by blood flow. When the heart pumps blood (systole), the amount of red increases, increasing the voltage, and when the blood is drained (diastole), the red decreases, decreasing the voltage, see figure 3.

The sensor sampled at a 200 Hz rate, and data acquisition implied the participation of 31 subjects (25 males and 6 females), with ages ranging from 22 to 55 years old.

Five different types of experiments were done with all the subjects. Such experiments

required as well the pronunciation of certain words or numbers, which were recorded with a microphone. This way, the database includes also wav files with the audios, and labeled files where the timestamps for every speech occurrence are annotated. Being so, it is possible to study the variation of the pulse knowing if there was speech or not in a certain period of time. All the experiments imply 30 seconds of pulse and audio recording, except for the fifth one, which took place during 1 minute. Unfortunately, the audio files for subjects S022 to S031 were corrupted, so the word labeling is not trustworthy.

The first experiment involved the subject saying two random credit card numbers of 16 digits each at a regular pace, with a longer pause between both numbers. The second one was a 30 seconds recording of the pulse and the audio without any speech coming from the subject. To continue with, the third experiment was similar to the first one, which implied the subject saying four random PIN numbers, where each PIN had six digits. Also, there was a longer pause between two consecutive PIN numbers. The fourth experiment used phonetically rich sentences for ASR, which the subject had to say at a regular pace and with a longer pause between sentences. Finally, the final experiment involved one minute of free speech, where the subjects would typically describe their environment.

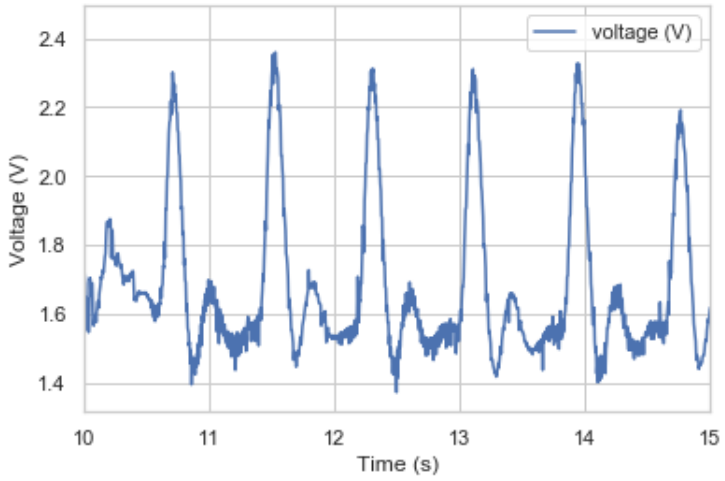

Figure 3: PPG measurement during 5 seconds from PulseID database.

3.1.1 Data obtainment and processing

PPG signal obtainment is done with a prototype specifically built for the *PulseID* dataset creation. This prototype uses a Raspberry Pi [12] as a computational module because of its small size, the free license and ease for working with other modules. The PPG sensor [11] is mounted on top of the Raspberry Pi, and a 10 bits Analog to Digital Converter (ADC) MCP3308 [13] is used to transform voltage variations into digital samples, at 200 Hz rate.

In order to synchronize up the PPG signal with the audio recordings from the subjects, a Python code is developed, which ensures a tolerant sampling rate deviation of $\mu = 13.32\,\mu s$ and $\sigma = 202.58\,\mu s$ per subject, see figure 4.

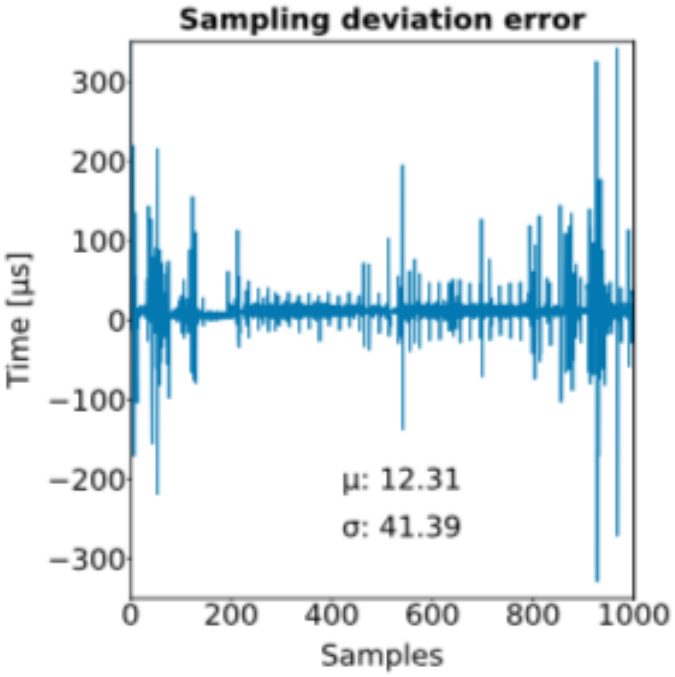

Figure 4: Sampling deviation error for a 5 second sample.

Once the PPG and the audio signals are recorded and synchronized, the latter is used for annotating the timestamps indicating the beginning and the end of every pronounced word. Due to the large effort that would suppose to perform such task by hand, labeling is done with an ASR trained using the Kaldi toolkit [14], as used in [15]. This ASR uses a single pass DNN system (4 hidden layers with 1024 neurons

each), with GMM pre-training, on top of filter-bank features. The GMM system uses discriminative feature transformations for GMM alignment. LDA and model-space adaptation with maximum likelihood linear transform (MLLT) are done on top of triphone acoustic models, with the objective of improving the separability between different acoustic classes in the feature space. Since the words that every speaker pronounced are previously known (except for the free speech case, which is not annotated), the ASR is fed with the words said. This way, force-alignment is done, since the ASR already knows which words have been said, and iteratively improves the alignment with the distribution that maximizes the probability.

Nevertheless, this system commits some errors while labeling the audio, specially when strong background noise happens, so still a manual cleaning has been done. Every audio file is loaded to WaveSurfer program, which allows for loading and aligning the file with the labels, as can be seen in figure 5. This way, imprecise labels have been correct by hand, adjusting them with WaveSurfer.

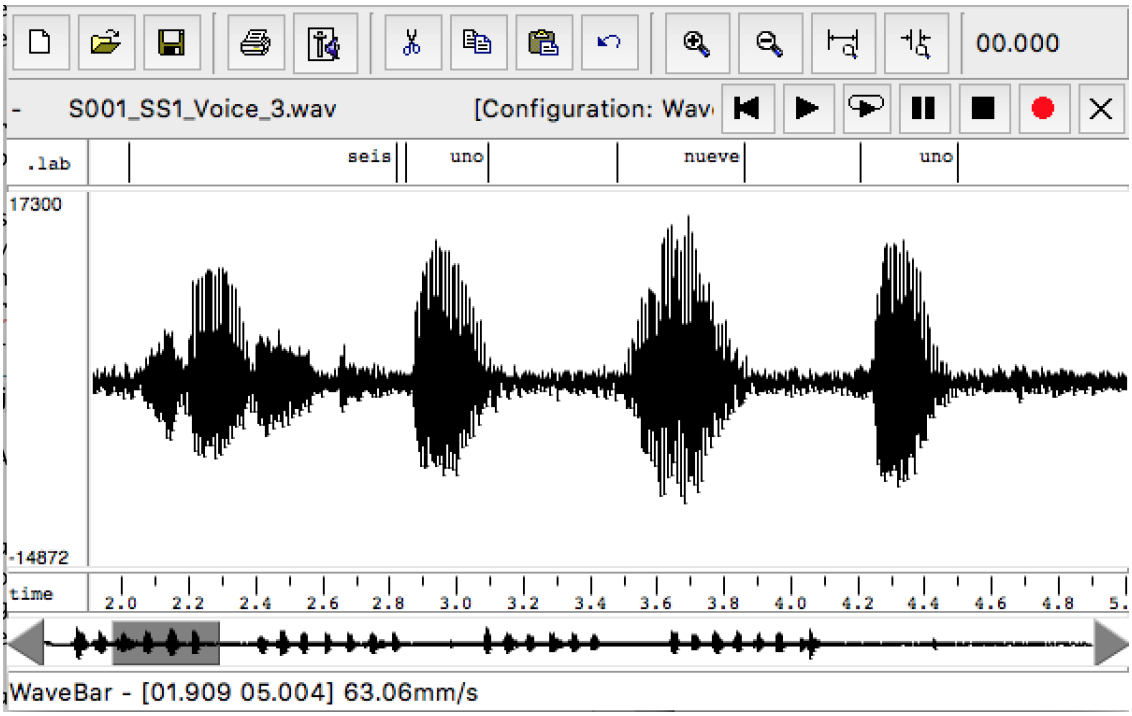

Figure 5: Audio file loaded in WaveSurfer, with the corresponding label file (.lab) aligned in the pane on top of it.

3.2 Biomarker Architecture

Typically, research techniques have focused on handcrafted extraction of relevant features from PPG and ECG signals for similar problems. However, the approach taken in this work is to use deep learning methods (Deep Neural Networks) that minimize the feature extraction effort. This way, architectures which are able to extract the biomarkers needed for classification are used here.

The code has been developed using *Python 3* programming language. For the implementation, training and testing of the Deep Neural Network architectures, *PyTorch* library has been used. *Pandas* and *Numpy* libraries have been used as well for data processing, and *Matplotlib* for plotting the results.

Two different Convolutional Neural Network architectures are used for speech events and gender classification tasks: the one from Luque et. al. work [2] and the VGG-like model used for noise detection in ECG signals [9]. Both architectures can be seen in figures 1 and 2. The latter is used in a straight-forward manner, but the first one is tuned up for the speech detection case. In other words, the initial kernel sizes for the three convolutional layers, which are L_1, L_2, $L_3 = 50, 30, 20$, are slightly modified in order to see which configuration is better with any of the kernel sizes $L_i \in$ [200, 180, 160, 140, 120, 100, 80, 60, 50, 40, 30, 20, 15, 12, 10, 8, 6, 5, 4, 3, 2].

Furthermore, a third CNN architecture is implemented for the gender classification task, which accepts bi-dimensional input from STFT spectrograms, see figure 6. This way it can be checked if representing the data in such way yields to better classification results, such spectrograms have proven to represent frequential information more clearly, as in [4] and [5] . This architecture contains four sequential blocks of convolutional layers, each one with 64, 128, 256 and 512 filters. Each block contains two sequential convolutional layers, each one of them with a kernel size of 3 and a stride of 1, followed by batch normalization and a ReLU activation function. After the convolutional blocks, a dense classifier of 1024 neurons is used.

Cross-entropy loss function is used with Stochastic Gradient Descent (SGD) opti-

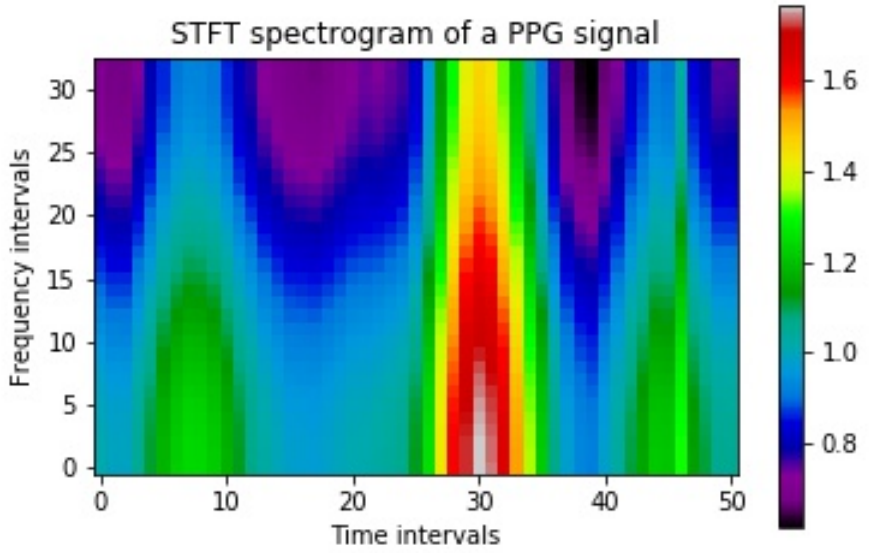

Figure 6: STFT spectrogram for a 1 second PPG signal, with a number of 64 FFT points, a 2% window stride and a 1% window size.

mizer. Batch size and learning rate are fine tuned for every architecture configuration to values allowing the model to safely decrease the loss and augment the accuracy. Particularly, a variant of SGD is implemented, which is called SGD with Restarts (SGDR). This technique uses *cosine annealing*, which consists of decreasing the learning rate in the form of half a cosine curve. When it reaches the minimum value, it is restarted to the original value, and decreases again in the same form, as can be seen in figure 7. This way, a high learning rate at the beginning allows for quicker training towards the minimum loss. However, since learning rate is decreased, it is also ensured that training does not fluctuate too much. Resetting the learning rate allows for a jump from the current local loss minima to another one, so the loss space is searched more thoroughly than with normal SGD.

The whole data set is split in training, validation and test sets, each one with a size of the 64, 16 and 20% of the original data set. The training set is used during forward passes and back-propagation of the gradients, for weights adjustment, and the validation set is evaluated to decide which epoch yielded the most accurate model.

The performance metric to make this decision can be decided in the execution configuration, but F1 macro average is typically used at all the experiments, preferred over accuracy, which might be misleading in cases where there is a class imbalance. Afterwards, this model is used to predict the classes of the test set, where many metrics are obtained, like AUC score, accuracy, precision, recall, F1-scores (weighted, macro...), etc. From these metrics, AUC and F1 weighted scores are the ones used to determine finally how good the model is, once again to account for class imbalances.

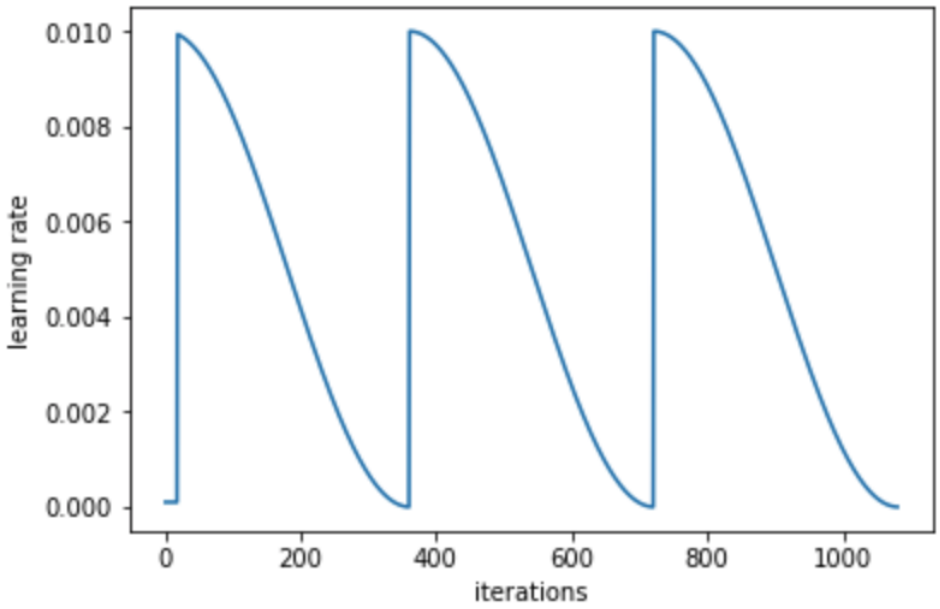

Figure 7: Cosine annealing applied on the learning rate with restarts.

3.2.1 Speech/Non-speech events

The recognition task of speech/non-speech events is done with deep learning methods, specifically using Convolutional Neural Networks (CNN) architectures, which have reported significant success in the task of image recognition. Being so, it seemed reasonable to use them as a starting point for an effective architecture.

Concretely, the same architecture as in [2] has been used, see figure 1, borrowed from [2] too.

Regarding data processing, the signals from experiments 1, 3 and 4 are taken, dis-

carding only the ones coming from noisy recordings (subjects from S022 to S031). Z-score normalization is done through all the 30 second samples, prior to further processing. Furthermore, most of the PPG signals have a certain amount of high frequency noise, which can be filtered with a 1D Gaussian filter. It is also studied if filtering this noise is beneficial or not for the classification task, since speech events are expected to cause certain anomalies in the signal.

For every signal recording of 30 seconds, smaller signal subsamples are extracted with a rolling window, which is passed between the first and the last timestamps where speech occurs, and every signal excerpt is labeled as "Speech". Therefore, an identical window is used from the last speech timestamp to the end of the recording, and it is labeled as "Non-Speech". Since there are short moments during speech where the speaker stops and breathes, if a "Speech" signal window contains more than a 2% of silence, it is discarded. This way it is ensured that the speaker was speaking the most of the time during signal samples labeled as "Speech".

See figure 8 for the visualization of two PPG signal samples which shall be fed to the CNN architecture: a "Speech" and a "Non-speech" sample.

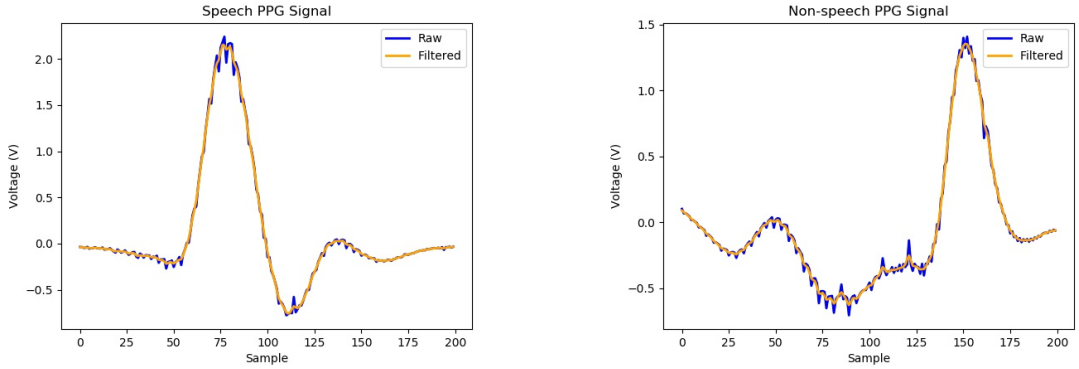

Figure 8: 1 second excerpt from a Speech PPG Signal (left) and another one from a Non-speech PPG signal (right). Raw signals are represented in blue, while signals filtered by a Gaussian filtered appear in orange.

Several sizes and strides are used for the rolling window, in order to see which one works the best. However, the most commonly used window size and the starting point in many experiments is the 1 second window, since this is the one used in

[2]. Changing the stride is interesting in order to see how signal overlapping affects the classifier accuracy. Being so, a desired overlapping percentage is introduced in the execution configuration, typically one of 0, 25, 50 or 75%, and the stride is automatically adapted to match that requirement. Signal overlapping is done after splitting the data set in training, validation and test set. Signals that are overlapped in a high percentage can be almost identical, and if such signals would be split in training and test sets, the CNN might classify correctly the one in the test set just because it is similar to the other one, which has already seen during training.

As an example, a "super-sample time window" is decided, let's say 2 seconds for instance. After having split all the data set in 2 second segments, these are split again in training, validation and test sets. Now, every 2 second signal is split once again according to the normal "time window" and the overlapping decided, which could be a 1 second window with 50% overlapping, for example. For this case, every 2 second super window would yield to 3 time windows of 1 second (first from 0 to 1 second, second from 0.5 to 1.5, and the last from 1 to 2). This way, every 1 second excerpt is fed into the CNN architecture as an input, and the network is trained with them. Afterwards, when evaluating the network with the validation and the test sets, the classifications are done with the super-sample time windows of 2 seconds, but taking into account the probabilities of the three 1 second samples extracted from each one of them. In other words, to decide if a super-sample 2 second signal is speech or not, the overlapped sub-samples extracted from it are passed into the model, and the output probabilities from each one of them are retained. Concretely, for every sub-sample the log-probability of being non-speech is subtracted from the probability of being speech. Thus, if the probability of being speech is higher, the score is positive, and if lower, the score is negative. Finally, the sum of every one of these scores is done, and the final label for the super-sample is decided regarding if the sum is positive (speech) or negative (non-speech), see equation 3.1.

$$\lambda = \sum_{i=0}^{N} log(a_i) - log(b_i) \tag{3.1}$$

where a and b are the probabilities of the sub-sample being speech or non-speech, respectively, and i is the index of the sub-sample. If $\lambda \geq 0$, the super-sample is classified as speech, and if $\lambda < 0$, otherwise.

However, if no overlapping is set, the super-sample time window and the sub-sample time window are equal, so no splitting is done and the decision on the class is done just by taking the maximum probability from the Log-Softmax output in the network.

3.2.2 Gender classification

As discussed in the Objectives section, the gender classification experiment could be related to pitch classification, because there is a clear correlation between a speaker's gender and his/her voice pitch. However, being able to classify the gender with the PPG signal does not necessarily mean that it is done because of the voice fundamental frequency leaking into the sensor. Other factors might be present in the signal, like the different morphology between men and women. Thus, the main method is to train and test a Neural Network architecture with speech only samples, silence only samples and a mix of them, separately. If it is found that the network using speech only samples performs significantly better than the other ones, it might be a clue that it is perceiving the fundamental frequency of the speaker's voice, enhancing its accuracy.

For this task, two different architectures are used: a variant of the PulseID network (PulseNet) with smaller kernel sizes, and the bi-dimensional CNN model using the STFT from the raw signal. Since the interest in this experiment relates to the frequential part of speech, it is interesting to see how the model using the STFT performs, because it reveals the frequency spectrum in such signal.

All the experiments for this task use 1 second window, and batch sizes and learning rates fine tuned to ensure convergence during training.

Chapter 4

Results

Find in this chapter the results for both experiments, speech/non-speech and gender classification, through all the different taken approaches: architectures, signal filtering, signal overlapping, time windows, etc.

4.1 Speech/Non-speech events

The results for speech and non-speech events classification are here presented. As mentioned in the Methods section, several approaches are tried, as a first exploration on which architectures and methods are better suited to detect speech from PPG signal.

4.1.1 PulseID architecture with 1D Gaussian Filter

First of all, a model with the same architecture used for speaker identification through PPG signal is trained for classifying PPG signals as speech or non-speech. This architecture is called PulseID or PulseNet architecture as an abbreviation. Two identical experiments are done here, except for the data processing. A 1D Gaussian Filter is passed in the the signals for the first variation of the experiment, whereas for the second no filter is used. The filter cleans the signal notably from high frequency noises. Both of the experiments are repeated a 100 times, each one of them

with the training (871 samples), validation (218 samples) and test sets (273 samples) shuffled, so all the scores are averaged across all the repetitions. Every signal passed to the model is 1 second long (200 samples) with no overlapping, with batch size set to 128 and learning rate set to 0.1. Since the classes in the test are a bit imbalanced (usually around 66% of non-speech and 33% of speech samples), the AUC and the F1-Weighted average scores are taken into account to evaluate the goodness of the model.

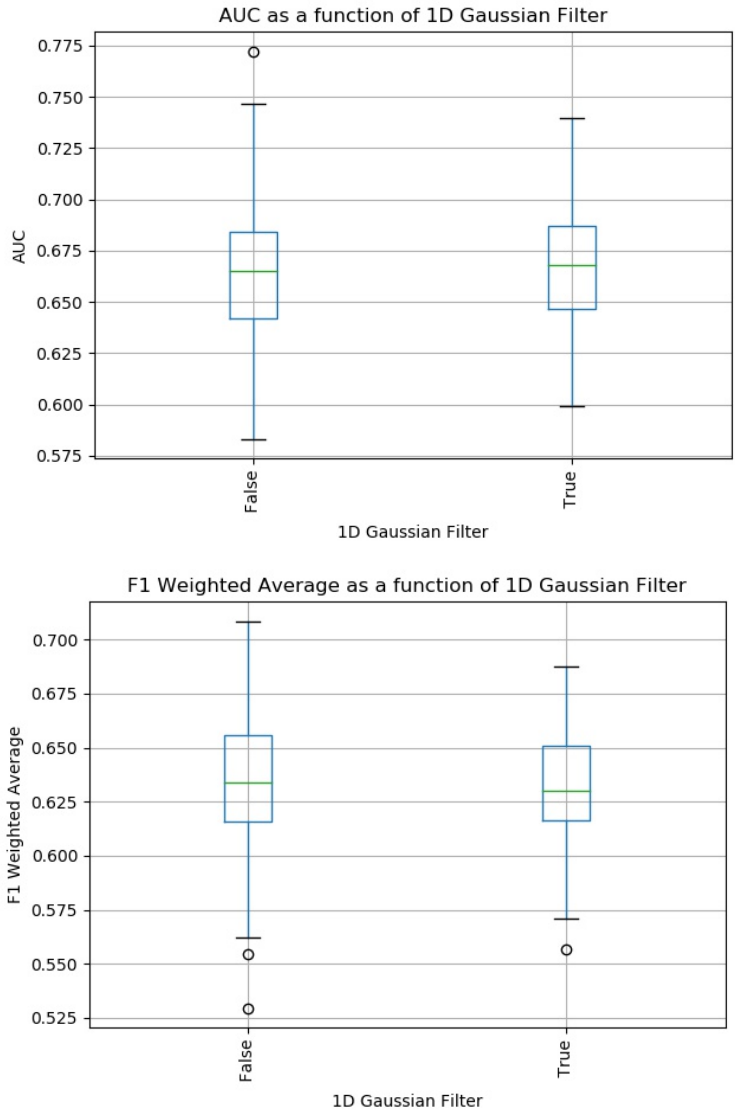

Figure 9: Boxplot representations of test set evaluation across 100 repetitions for signals processed with and without a 1D Gaussian Filter, AUC (top) and F1-Weighted Average (bottom) scores.

This first exploration shows that the system is able to classify speech and non-speech

PPG samples with a relatively good accuracy for both cases, fairly over random guessing, see figure 9. For the case without 1D Gaussian Filter, a $66.6 \pm 0.3\%$ mean AUC and a $63.5 \pm 0.3\%$ mean weighted average F1-score are found. For the case with the filter, the mean AUC is $66.8 \pm 0.3\%$ and the mean weighted average F1-score is $63.3 \pm 0.3\%$. Therefore, both results are really similar and within the error ranges, so the implication of the filter is not significant. On the other hand, from all the experiments' repetitions, the best performing model has scored a 77.2% AUC and a 70.9% weighted F1-score, see figures 10, 11, and table 1.

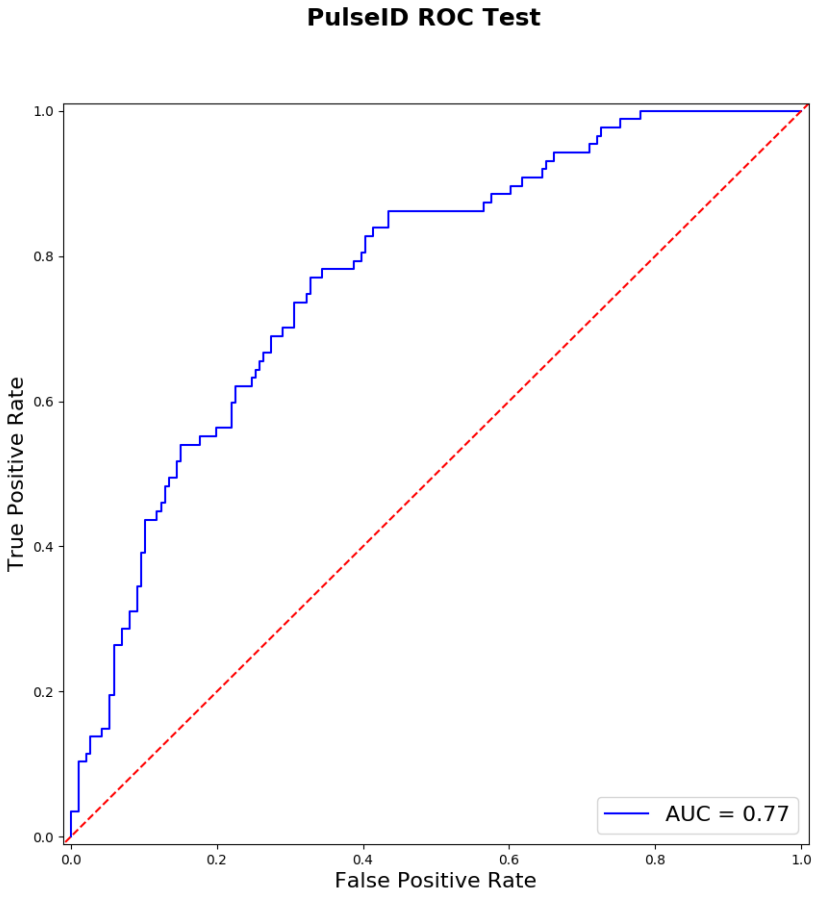

Figure 10: Speech and non-speech classification AUC on the test set, using the best PulseID model from all the experiments' repetitions.

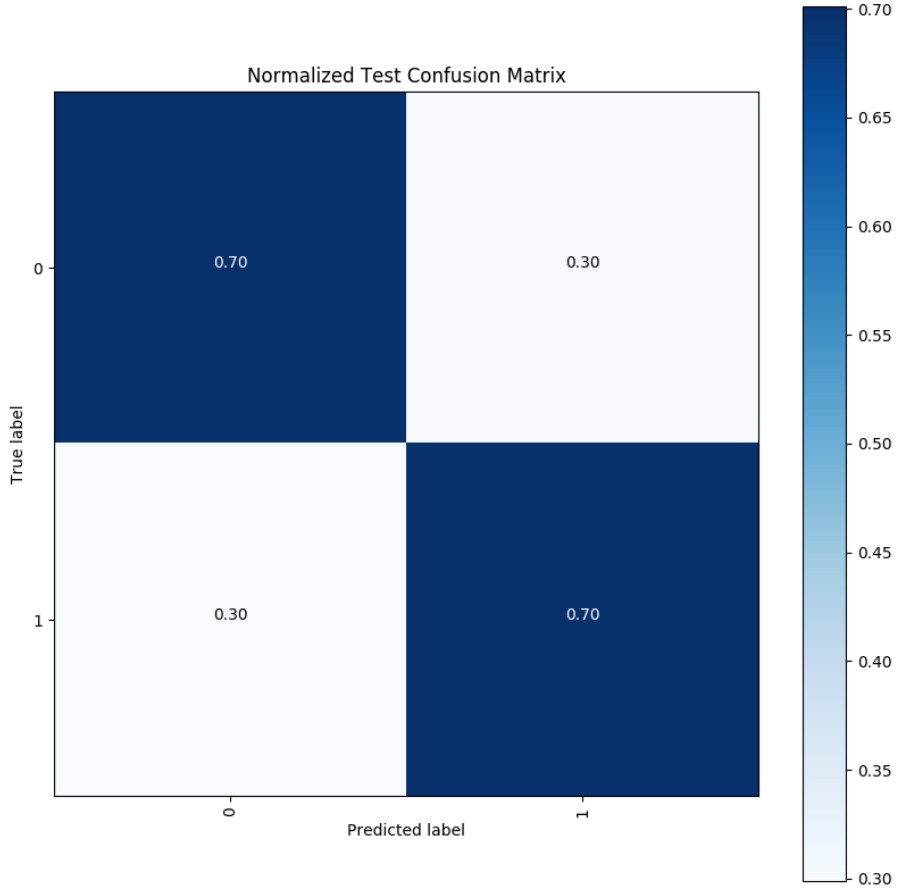

Figure 11: Normalized Test Confusion Matrix, using the best PulseID model from all the experiments' repetitions.

Table 1: Speech/Non-speech classification results table.

	Precision	**Recall**	**F1-score**	**Support**
Non-Speech	0.83	0.70	0.76	186
Speech	0.52	0.70	0.60	87
Micro Avg	0.70	0.70	0.70	273
Macro Avg	0.68	0.70	0.68	273
Weighted Avg	0.73	0.70	0.71	273

4.1.2 Inverted VGG16, PulseID, PulseID Variant and bi-dimensional CNN architectures

To continue with, the results for the exploration on four different architectures is done, for the 1 second time window without overlapping case. The same conditions as in the previous subsection are used (every different configuration is repeated 100 times with shuffled data partitions, to obtain relevant statistical information), except for batch sizes and learning rates, which are adjusted for every architecture to ensure convergence during training. As mentioned in the Methods section, the proposed architectures are: the Inverted VGG16 used in [9], PulseID model as described in [2], a variant of PulseID with smaller kernel sizes (15, 8 and 2) and a shorter VGG-like CNN adapted for bi-dimensional input, which uses the STFT of the signal instead of the raw signal in 1D. The intuition behind using a PulseID model with smaller kernel sizes is to explore if the correlations in the signal are in a shorter scale than in the speaker identification case.

As can be seen in figure 12, the best performing architecture is the variant of the PulseID network (PulseNet-Var), with a $67.6 \pm 0.3\%$ mean AUC and a $64.0 \pm 0.3\%$ mean F1 Weighted Average score, slightly above the normal PulseID model (PulseNet), which scored a $66.8 \pm 0.3\%$ mean AUC and a $63.3 \pm 0.3\%$ mean F1 Weighted Average score, as seen in the previous section.

The other two models have a significantly worse performance, with a $64.6 \pm 0.3\%$ mean AUC and a $62.6 \pm 0.3\%$ mean F1 Weighted Average score for the Inverted VGG16 (VGG16_Inverse) and a $61.7 \pm 0.3\%$ mean AUC and a $60.4 \pm 0.3\%$ mean F1 Weighted Average score for the bi-dimensional CNN (CNN_2D).

Being so, on a first glance it seems that for the chosen time window of 1 second, the best performing network is the PulseID variant, which seems to benefit of smaller kernel sizes to find correlations in the signal. The Inverted VGG16 inspired in the noise detection in ECGs model has a fairly good performance, but not as good, and finally the bi-dimensional CNN processing STFT seems to have fewer success classifying the samples. On and on, the best approach for this first exploration

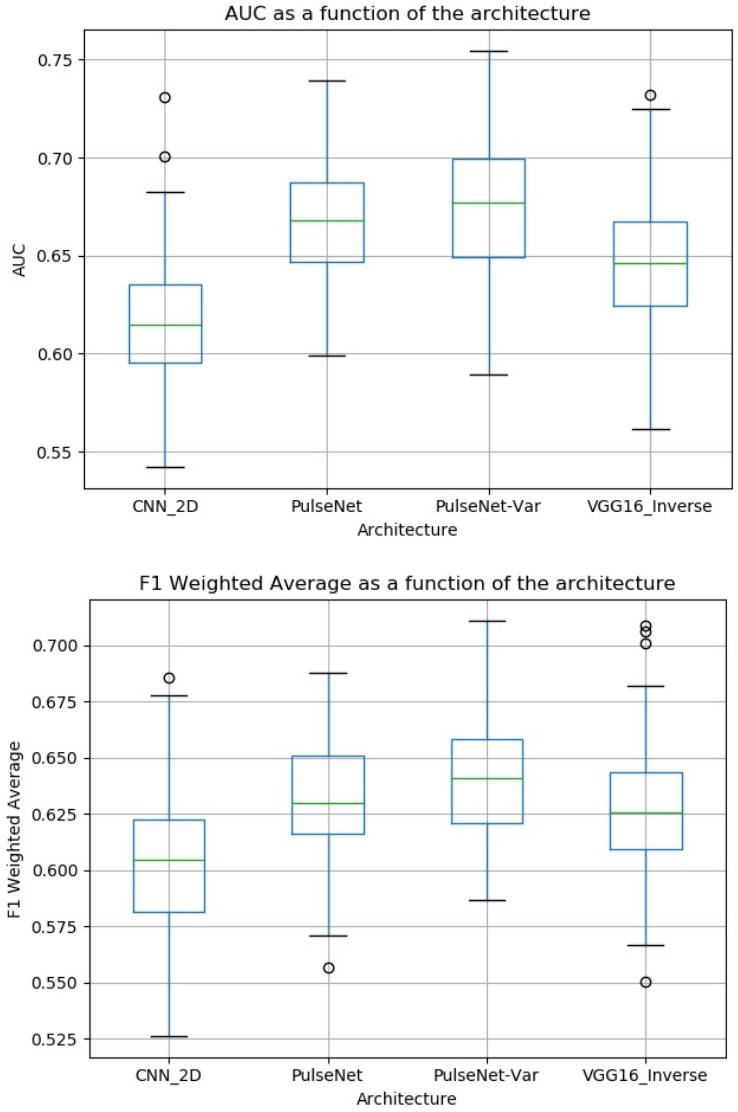

Figure 12: Boxplot representations of test set evaluation across 100 repetitions for signals processed with four different architectures, AUC (top) and F1-Weighted Average (bottom) scores.

seems to be using PulseNet models, with slight variations.

4.1.3 Overlapping or not overlapping

As mentioned in the Methods section, it is possible to train and test with signals overlapped with each other. The decision about the label of a signal is decided by the combination of the probability outcome from all its overlapped sub-signals, which have been passed to the model. After a quick scan, it seemed that 70% overlapping

yielded the best performances, so the results from the following experiments are given by that same percentage of overlapping.

First of all, let's see how the scores change for the normal PulseNet architecture, when overlapping is used. Here, it is compared the previously known result of classifying a 1 second time window signal with no overlapping, against classifying the same signal size but using the probabilities from overlapped sub-signals of 0.3, 0.4, 0.5 and 0.6 seconds.

Results, as can be seen in figure 14, show that evaluating a 1 second time window with 70% overlapped sub-samples of 0.4 seconds yields a better result than testing directly over the whole 1 second sample, without overlapping. As seen in the previous subsection, a PulseNet architecture (1 s baseline) shows a $66.8 \pm 0.3\%$ mean AUC and a $63.3 \pm 0.3\%$ mean F1 Weighted Average score, whereas classifying 1 second samples with overlapped 0.4 s sub-samples gives a $67.6 \pm 0.3\%$ mean AUC and a $64.7 \pm 0.3\%$ mean F1 Weighted Average score. This is at the moment the best result, similar to the one from the PulseNet variant, but with a better F1 Weighted Average score.

On the other hand, let's check how sub-sample overlapping affects the prediction accuracy of the PulseNet variant, which is the best performing model for whole 1 second samples with no overlapping. This time, the subsignals tried have time windows of 0.15, 0.20, 0.30, 0.35, 0.40 and 0.45 seconds. Since the kernel sizes are smaller in this architecture, it is worth checking the accuracy for smaller sub-samples also.

The boxplots in figure 14 show that no overlapping sub-samples yield better results than classifying the signal with the whole time window without overlapping ($67.6 \pm 0.3\%$ mean AUC, $64.0 \pm 0.3\%$ mean F1-score), for the PulseNet variant. As with the normal PulseNet model, the best performing sub-sample time window is 0.4 s, but with a slightly worse performance in this case ($66.7 \pm 0.3\%$ mean AUC, $64.0 \pm 0.3\%$ mean F1-score).

Being so, it seems like PulseNet variant with smaller kernel sizes is able to do a better

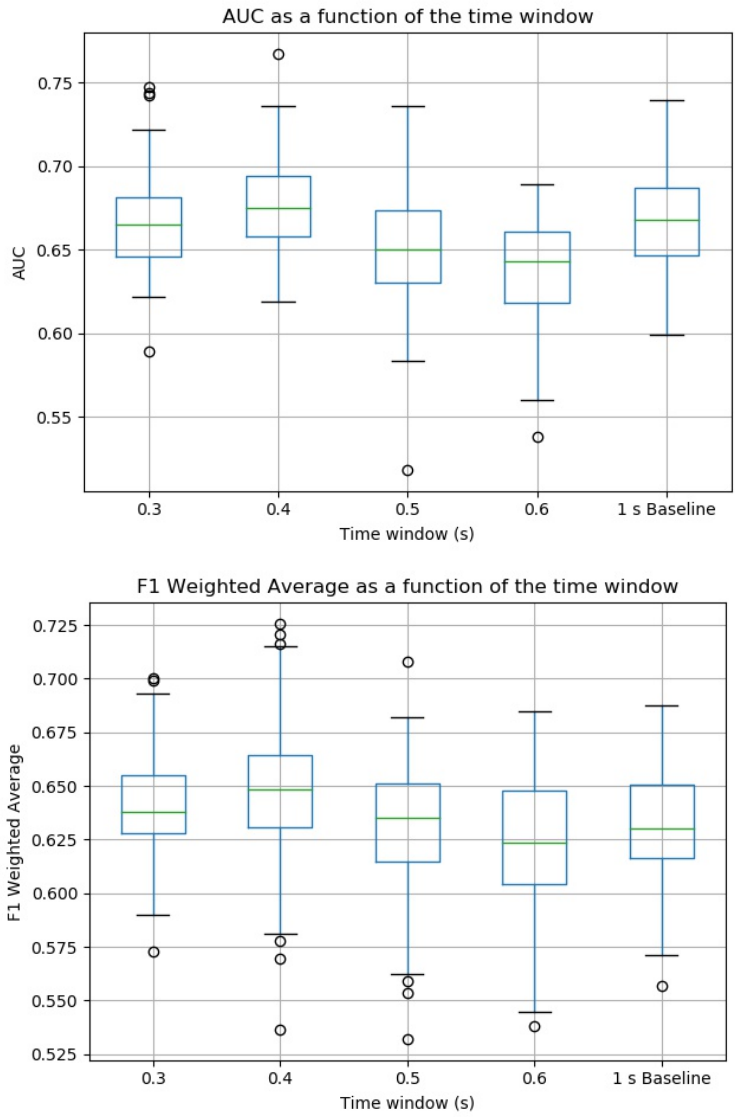

Figure 13: Boxplot representations of test set evaluation across 100 repetitions for signals processed with overlapping of different time windows, which are combined to classifying a 1 second super-sample, with the PulseNet model. The rightmost boxplot (1 s Baseline) is the one corresponding to evaluating a 1 second sample as a whole, without overlapped sub-samples. AUC (top) and F1-Weighted Average (bottom) scores.

classification without overlapping samples, whereas regular PulseNet with bigger kernel sizes benefits of a better performance when overlapping samples are used. A possible explanation about such finding can be found later on in the Discussion section.

Other super-sample time windows have been tried, instead of the 1 second time

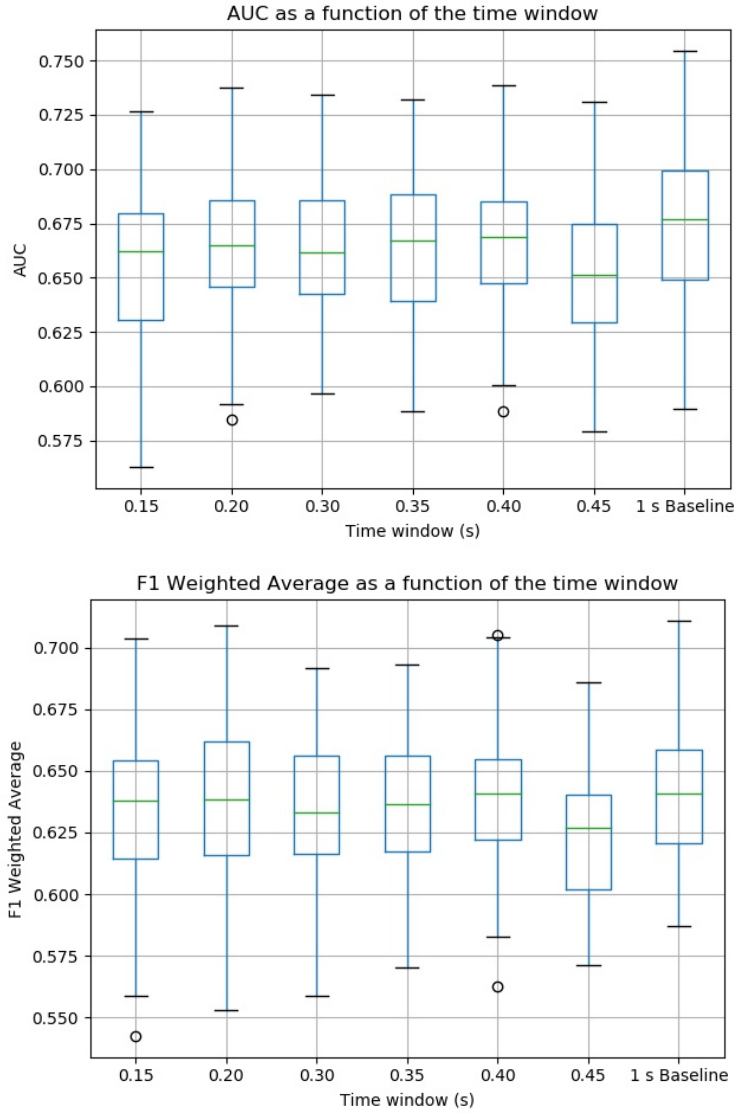

Figure 14: Boxplot representations of test set evaluation across 100 repetitions for signals processed with overlapping of different time windows, which are combined to classifying a 1 second super-sample, with the PulseNet variant model. The rightmost boxplot (1 s Baseline) is the one corresponding to evaluating a 1 second sample as a whole, without overlapped sub-samples. AUC (top) and F1-Weighted Average (bottom) scores.

window, but results are fairly worse. For example, using a 2 second time window split in 70% overlapping 1 second sub-samples yields a $64.4 \pm 0.3\%$ mean AUC for the regular PulseNet model, and a $63.7 \pm 0.3\%$ mean AUC for the PulseNet variant model. Using a 0.5 second sub-sample window for this late model yields a better $65.3 \pm 0.3\%$ mean AUC score, but still worse than the best score of $67.6 \pm 0.3\%$ given by the non-overlapping 1 second windows, as previously seen.

4.1.4 Deeper exploration on PulseNet variants

As previously mentioned, a quick exploration on PulseNet kernel sizes showed that a [15, 8, 2] configuration yielded better results than the original [50, 30, 20] one. However, still a deeper exploration on several configurations has been done, to see which maximum AUC could be achieved by tweaking the kernel sizes only. 150 different combinations have been tried, the best 5 of them are presented in table 2. As can be seen, a new best mean AUC score is found with a $68.2 \pm 0.3\%$, for the [50, 10, 4] configuration. It seems that is beneficial to maintain the largest filter with a size of 50, as in the original PulseNet, but then using smaller sizes for the other two filters. It seems like using an 80 size filter instead is also fairly good. The exception to the small filters pattern is the [80, 60, 20], which using greater filter than the original architectures, achieves good results.

Table 2: AUC as a function of the best PulseNet kernel sizes L_1, L_2, L_3.

$\mathbf{L_1}$	$\mathbf{L_2}$	$\mathbf{L_3}$	**AUC**
50	10	4	68.2 ±0.3
50	15	3	68.0 ±0.3
80	6	2	67.6 ±0.3
80	60	20	67.6 ±0.3
80	6	4	67.6 ±0.3

Table 3: AUC as a function of the worst PulseNet kernel sizes L_1, L_2, L_3.

$\mathbf{L_1}$	$\mathbf{L_2}$	$\mathbf{L_3}$	**AUC**
160	140	8	63.3 ±0.3
200	20	2	62.9 ±0.3
160	140	120	62.8 ±0.3
200	140	50	62.5 ±0.3
200	160	120	61.7 ±0.3

However, let's examine how are the configurations yielding the worst AUCs, in 3. It looks as bigger kernel sizes offer a poorer performance, with the lowest AUC being

$61.7 \pm 0.3\%$, yielded by the [200, 160, 120] configuration. Therefore, this is a clue that the influence of speech in 1 second time windows of PPG signal might be given by short fluctuations that are best captured by smaller kernel sizes.

4.2 Gender classification

Gender classification task can be approximated as a speaker's pitch classification task, since typically a male's voice is lower pitched than a female's one. However, as discussed in the Methods section, a Neural Network architecture might be able to classify the gender through PPG because of factors others than the pitch, like simple morphological variations between genders. Thus, it is interesting to see how the presence of speech during a PPG signal affects the prediction of the model regarding the genre. If the PPG signal contains relevant information about the speech's fundamental frequency, then the classifier would have an additional clue about the sample's corresponding genre, then yielding to a better performance. Otherwise, if this information is not filtered in the PPG, no difference should be noticed. All the experiments carried here are done with 1 second PPG samples, and no Z-score normalization is done, since it seemed to yield worse results.

4.2.1 PulseNet variant

Let's examine the effect of training and testing the PulseNet variant with three different data sets: one with speech samples only, another with silence samples only and a third with a mix of them. Since a mix of them would have more data points, the data sets are shuffled and forced to have a maximum number of 482 samples, which is the number of speech samples, the class with the lowest number of samples.

The results, as can be seen in figure 15, are quite surprising. First of all, it is shown that the architecture is quite good classifying the gender through PPG signal, achieving a $85.0 \pm 0.3\%$ mean AUC in the case where only silence samples are used. However, contrary to the hypothesis, it seems that for the cases where speech samples are used, the performance drops. The mean AUC is $80.2 \pm 0.3\%$ for the speech-only samples and $83.4 \pm 0.3\%$ when they are mixed with silence samples. The differences

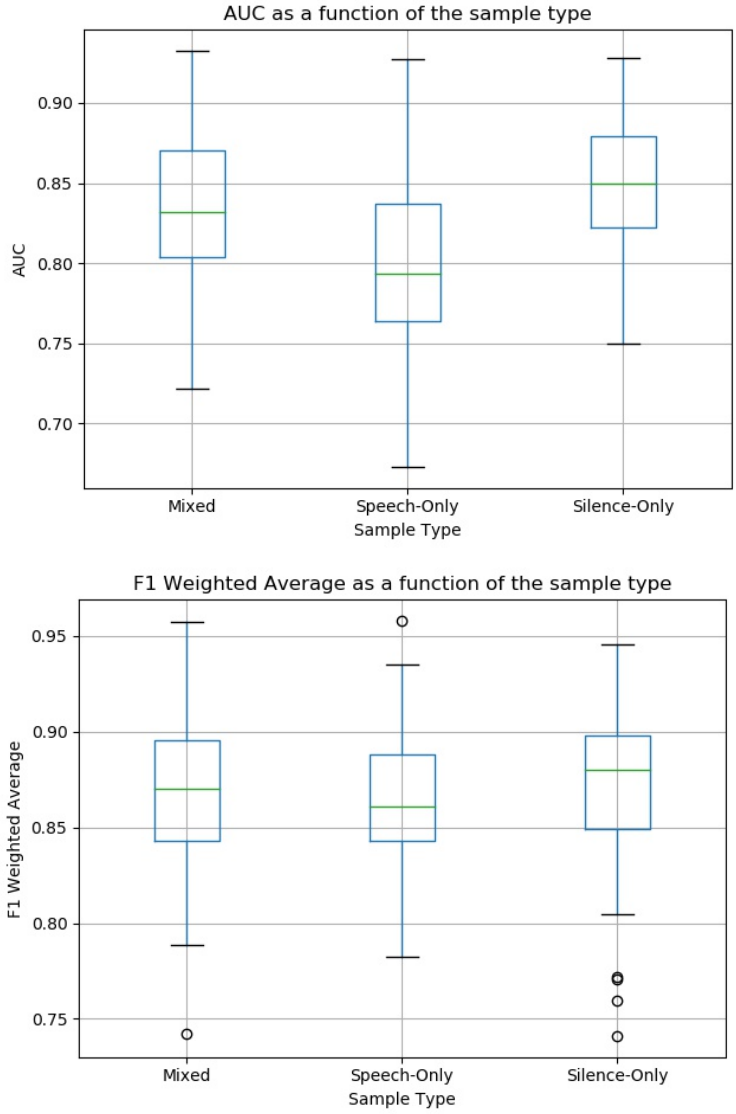

Figure 15: Boxplot representations of test set evaluation across 100 repetitions for gender classification task, as a function of the sample type, with the PulseNet variant model. AUC (top) and F1-Weighted Average (bottom) scores.

are significant, which gives an idea that speech samples might be just noisier, giving the model a harder time for classifying. Being so, it does not seem that there is information related to voice pitch in the PPG signal. At least, if such information is really present in the signal, the PulseNet variant model is not able to capture it.

4.2.2 Bi-dimensional CNN

Since it is expected to find some leak from the fundamental frequency of the speaker in the PPG signal, and the PulseNet has not had much success perceiving it, the bi-dimensional CNN is tried for the same task. As previously explained, the STFT from the signal is passed into the network, which should aid the model to find frequential information, since it is more clearly represented in the spectrogram.

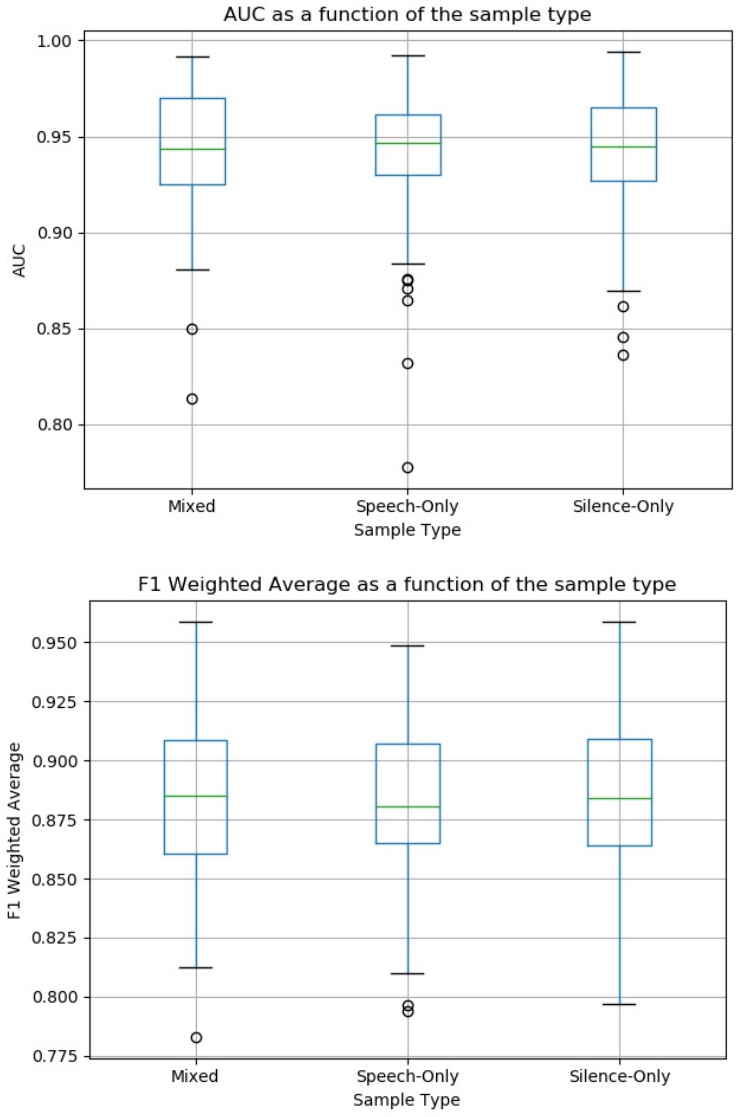

Figure 16: Boxplot representations of test set evaluation across 100 repetitions for gender classification task, as a function of the sample type, with the bi-dimensional CNN model. AUC (top) and F1-Weighted Average (bottom) scores.

The first noticeable result in figure 16 is that the performance of the model is much

better than the PulseNet variant, with a $94.2 \pm 0.3\%$ mean AUC for the best case, which implies a mixture of speech and non-speech samples. For the speech only and silence only sample cases, the AUC is practically the same, a $94.1 \pm 0.3\%$. Being so, for this architecture, the differences are not significant, actually with a very slight improvement for mixed samples. The presence or not of speech in the PPG does not cause a significant difference in the performance. This means that if speech only causes noise in the sample, the model is not distracted by it, which is an improvement respect the PulseNet variant. On the other hand, if speech is not just noise, and carries frequential information, it does not contribute to the model being better. It is possible that the model is looking for other features that have a greater weight on determining if the subject is male or female, without giving significant attention to the fundamental frequency from the voice pitch.

4.2.3 Bi-dimensional CNN with larger mixed data set

Just as a side experiment, not directly related to prosody, the bi-dimensional network is trained without limiting the number of samples, only to see how good it can be with more data. Thus, instead of using a limited 482 samples data set, 1362 samples are used. The classification performance is very good, achieving a mean AUC of $97.58 \pm 0.09\%$ and a mean weighted average F1-score of $93.2 \pm 0.1\%$. Notice in figure 17 how the AUC is significantly higher than training with fewer samples, and how the dispersion in the results is tighter.

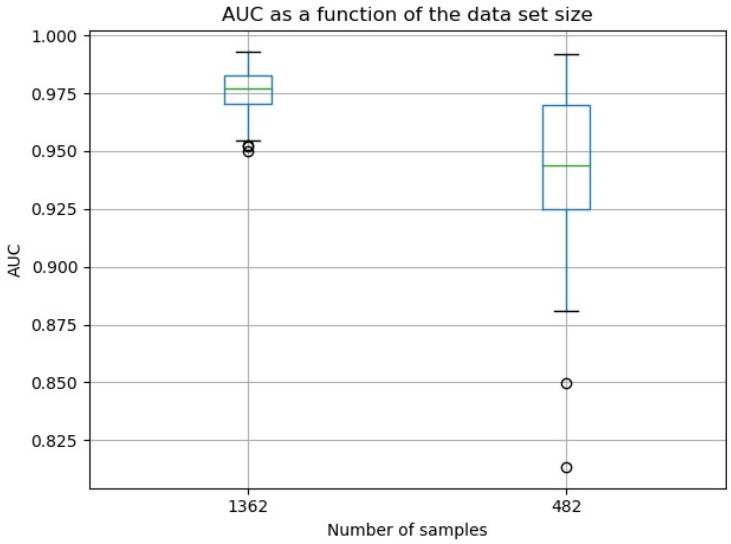

Figure 17: Boxplot comparison between the AUCs for the bi-dimensional CNN trained and test with 482 and 1362 samples.

Chapter 5

Discussion and Conclusions

Having presented and seen all the results, a discussion taking into account all of them is done in this section. From such discussion, the final conclusions on the correlation of speech and PPG signal are given.

5.1 Discussion

Before starting a deeper discussion on the results, let's summarize in table 4 the most relevant scores found in the previous section.

Table 4: Speech/Non-speech final results table. All results have 1 second super-sample time windows.

Arc	**Sample Size (s)**	**AUC**	**F1-Score**
PulseNet-Var1	1	**68.2 ± 0.3**	63.9 ±0.3
PulseNet-Var2	1	67.6 ±0.3	64.0 ±0.3
PulseNet	1	66.8 ±0.3	63.3 ±0.3
VGG16-Inv	1	64.6 ±0.3	62.6 ±0.3
CNN-2D	1	61.7 ±0.3	60.4 ±0.3
PulseNet-Var	0.4	66.7 ±0.3	64.0 ±0.3
PulseNet	0.4	67.6 ±0.3	**64.7 ± 0.3**

All four architectures are able to distinguish speech from silence through PPG sig-

nals, not with a high accuracy, but indeed way over the random baseline, with all AUC and F1-Weighted over 60%, averaged in runs of 100 experiments. Therefore, this first exploratory study shows that there is a correlation between speech and non-speech events with PPG signal. The data set is relatively small, which is a drawback for deep learning architectures that usually required larger amounts of data, specially for cases like the one in this work, where it is not trivial by sight if a sample belongs to one class or another. Furthermore, the data set is quite noisy, even for non-speech samples, which can be seen in figures 3 and 8. In this last figure, it can be seen how the shown non-speech PPG signal is noisier than the speech one. For a 1 second time window, it is expected that the influence from speech in the signal might be subtle, which is hardly seen if the overall noise is certainly high. Still, the models find a way to achieve relatively good performance in speech and non-speech events classification.

Comparing the models, it seems like the PulseNet and its variants with smaller kernel sizes (PulseNet-Var1 with [50,10,4] configuration and PulseNet-Var2 with [15,8,2]) are the best performing architectures. The PulseNet architecture was previously used for finding correlations between PPG signals and speaker IDs, so it is probably best suited by default to deal with PPG signals. However, for the case of finding speech within such signals, it seems like it benefits from smaller kernel sizes, which find correlations in smaller sample windows. This way, the PulseNet variant, with kernel sizes of [50, 10, 4], as opposed to the original with [50, 30, 20], achieves the best performance without signal overlapping, with a $68.2 \pm 0.3\%$ mean AUC and a $63.9 \pm 0.3\%$ mean F1 Weighted Average score. Nevertheless, it is shown that the original PulseNet model is able to practically match this performance, by using overlapping of 0.4 second signals over a total 1 second sample, with a $67.6 \pm 0.3\%$ mean AUC and a $64.7 \pm 0.3\%$ mean F1 Weighted Average score. Other architectures like the bi-dimensional CNN and the inverted VGG16 have less success with this problem.

On the other hand, experiments with higher time windows, like 2 seconds, show worse results. The PulseNet architecture was also designed to cover 1 second sam-

ples, so it might not be so well fitted for largest ones. Nevertheless, it is known that the effect of breathing is reflected in the PPG signal in larger windows. The act of speaking causes irregularities during breathing, so this could be probably seen by a well-tuned architecture. However, this option has not been thoroughly tried in this work because using larger samples would hugely reduce the number of data points, and the data set for 1 second samples is already relatively small, even though sufficient. An interesting work for the future would be to increment the data set and perform such experiments. Actually, it is hypothesized that an ensemble model with architectures covering large and small time windows would increase the accuracy of the system. Such architectures would take into account fine-grained fluctuations from speech in the PPG signal, as well as the longer fluctuations caused by irregular breathing. Also, it would be interesting to see how an RNN or LSTM like architecture would work for this case, which might be able to find these sequential correlations.

All in all, even though it is proved that there is a correlation between speech and non-speech events with PPG signal, it is not clear which is the nature of such correlation. Does speech cause a drop in oxygen which causes a fluctuation in the analog voltage? Or is it just some noise from the acoustic vibration captured by the sensor? Would this noise carry frequential information from the speaker's voice? These questions remain unanswered by this sole experiment, which leads to the first exploration on prosody through PPG signal, done by gender classification with it, which is an approximation of classifying the pitch from the speaker's voice. Let's check the gender classification summarized results in table 5.

It is seen that the PulseNet variant classifies worse when only speech samples are used for training and testing, than when silence samples are given instead. Being so, it seems that such model is good classifying the gender without need of speech happening, and when it does, it is confused and has a harder time performing this task. This is contrary to the belief that speech samples might carry some information about the fundamental frequency of the speaker's voice, which could help distinguishing men from women. Therefore, it looks like the fluctuations caused

Table 5: Gender classification results table.

Arc	Sample Type	AUC	F1-Score
PulseNet-Var	Speech	80.2 ±0.3	86.5 ±0.3
PulseNet-Var	Silence	85.0 ±0.3	87.2 ±0.3
PulseNet-Var	Mix	83.4 ±0.3	86.9 ±0.3
CNN-2D	Speech	94.1 ±0.3	88.4 ±0.3
CNN-2D	Silence	94.1 ±0.3	**88.6 ± 0.3**
CNN-2D	Mix	**94.2 ± 0.3**	88.5 ±0.3

by speech do not contain relevant prosodic information, and might be just noise. However, a second architecture is tried, which processes the bi-dimensional STFT from the raw PPG signal, which is a clearer representation of the frequencies in it. This architecture shows an outstanding performance, with a $94.2 \pm 0.3\%$ mean AUC for mixed samples. As opposed to the previous model, for this one the presence or not of speech samples does not cause a significant variation in performance. This means that this model might be able to see the fundamental frequency from the speaker's voice, but still not greatly benefit from it, since other factors in the STFT spectrogram might have a stronger ponderation towards the classifier's decision. On the other hand, if speech is just noise in the STFT without relevant frequential information, then this model could be just good enough to not care about it and maintain a good classification rate without perturbations. By the end of this first pitch experiment, it is still unclear what is the nature of the speech correlation with PPG signal, and if relevant prosodic information can be extracted from it. Further experiments should be done, in order to determine the pitch of the speaker, independently of the gender, which is easily perceived by factors not related to speech, like possibly the heart morphology of men and women. Since the data set contains experiments with credit card and PIN numbers, and vocabulary rich sentences, an interesting experiment to try in the future would be to determine if a number sequence or a normal sentence is being said. Both of them have usually different prosodies. However, these experiments would need a larger data set, since a reasonable minimum time window would be between 2 and 5 seconds, which for

the current data set size would lead to fewer samples.

As a side consequence of this experiment, not related to the purpose of this work, but worth mentioning, it has been found that training this bi-dimensional CNN model with more samples leads to excellent results in gender classification. A mean AUC of $97.58 \pm 0.09\%$ and a mean weighted average F1-score of $93.2 \pm 0.1\%$ are retrieved. Up to the author's knowledge, this is the first attempt to perform such task with deep learning methods.

Finally, to explore the potential of PPGs for word recognition, some VGG-like models have been trained for recognizing digits from 0 to 9, from just a PPG signal. Several experiments have been done, varying the configuration filters and max-pooling operations, achieving a $12.0 \pm 0.2\%$ mean accuracy over them, not too far away from randomness (10%). However, a particular model with 7 layers achieved a 18% accuracy, with a 0.48 second time window, see figure 18 and table 6.

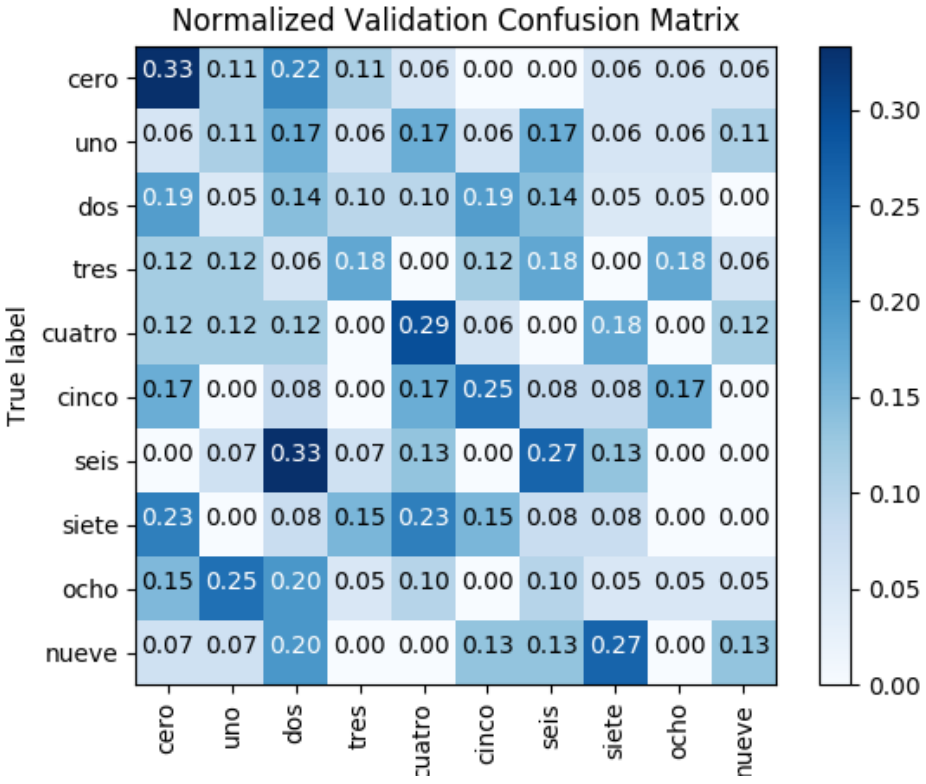

Figure 18: Confusion Matrix for the best scoring VGG-like model in Digits Recognition.

Table 6: Digits recognition results table.

	Precision	**Recall**	**F1-score**	**Support**
0	0.25	0.33	0.29	18
1	0.12	0.11	0.12	18
2	0.11	0.14	0.12	21
3	0.25	0.18	0.21	17
4	0.25	0.29	0.27	17
5	0.20	0.25	0.22	12
6	0.21	0.27	0.24	15
7	0.07	0.08	0.07	13
8	0.11	0.05	0.07	20
9	0.22	0.13	0.17	15
Micro Avg	0.18	0.18	0.18	166
Macro Avg	0.18	0.18	0.18	166
Weighted Avg	0.18	0.18	0.17	166

The configuration of the layers regarding the number of filters is the following: [2,2,M,4,M,4,M], where 'M' means that there is a max-pooling operation. Even though these results cannot be considered as conclusive, since it would be desired to obtain a consistent accuracy of at least 20%, they settle a starting point for further investigation on the possibility of word recognition with PPG signal.

5.2 Conclusions

A first exploration on the correlation between speech/non-speech events and PPG signal has been done, proposing and comparing four different CNN architectures for such task. All these architectures, trained and tested with the *PulseID* data set, show mean AUC scores in the range of [61.7%, 68.2%] over runs of 100 experiments, fairly above randomness, proving the correlation between speech and PPG signals and paving the way to further research on the improvement of speech detection models. The best performing model in an individual experiment achieves a 77.2%

AUC.

However, the clear nature of the fluctuations in PPG signals caused by speech is still pending to be clarified. It is not certain if these carry prosody markers or are just noise. A first attempt on performing pitch classification between high and low voice pitch is done, but the network seems to be simply classifying male or female, because it is not particularly benefiting from samples carrying speech fluctuations. The same 94.1±0.3% mean AUC is retrieved from experiments using speech only and silence only samples, so it is unclear if besides morphological information present in the heart beat, pitch information from the speaker's voice fundamental frequency is present in it. As a side note, apart from the purpose of this work, a 97.3±0.09% mean AUC is achieved when feeding the best performing 2D CNN gender classification model with more samples, an outstanding result.

To conclude with, even though the presence of prosody markers is still unclear, this work shows the possibilities of extracting speech information from PPG signal with end-to-end CNN architectures, achieving successful results on speech detection. This first exploration opens up for further research on this topic, which would allow the creation of new biometric applications, specially if the exact nature of speech fluctuations in PPG signal is discovered.

List of Figures

List of Tables

Bibliography

[1] Segura, C., Balcells, D., Umbert, M., Arias, J. & Luque, J. Automatic speech feature learning for continuous prediction of customer satisfaction in contact center phone calls. In *LNCS: Advances in Speech and Language Technologies for Iberian Languages*, 255–265 (2016).

[2] Luque, J. *et al.* End-to-end photoplethysmography (ppg) based biometric authentication by using convolutional neural networks. In *2018 26th European Signal Processing Conference (EUSIPCO)*, 538–542 (2018).

[3] Moon, S.-J. & Lindblom, B. Two experiments on oxygen consumption during speech production: vocal effort and speaking tempo. In *Proceedings of the 15th International Congress of Phonetic Sciences*, 3129–3132 (2003).

[4] Michalevsky, Y., Boneh, D. & Nakibly, G. Gyrophone: Recognizing speech from gyroscope signals. In *23rd USENIX Security Symposium (USENIX Security 14)*, 1053–1067 (USENIX Association, San Diego, CA, 2014). URL `https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/michalevsky`.

[5] Matic, A., Osmani, V. & Mayora, O. Speech activity detection using accelerometer. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference* **2012**, 2112–5 (2012).

[6] Korkmaz, O. E., Aydemir, O. & Öztürk, M. Detection of smoking, gender and starvation - satiety using photoplethysmogram signals. In *2017 25th Signal Processing and Communications Applications Conference (SIU)*, 1–4 (2017).

[7] Gu, Y. Y., Zhang, Y. & Zhang, Y. T. 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.*, 13–14 (2003).

[8] Spachos, P., Jiexin Gao & Hatzinakos, D. Feasibility study of photoplethysmographic signals for biometric identification. In *2011 17th International Conference on Digital Signal Processing (DSP)*, 1–5 (2011).

[9] John, J. N., Galloway, C. & Valys, A. Deep convolutional neural networks for noise detection in ecgs. *arXiv preprint arXiv:1810.04122* (2018).

[10] Simonyan, K. & Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[11] Murphy, J. & Gitman, Y. PulseSensor Open Hardware . `http://pulsesensor.com/` (2017). [Online; accessed 19-October-2017].

[12] Upton, E. Raspberry Open Hardware . `https://www.raspberrypi.org/` (2018). [Online; accessed 10-June-2018].

[13] Microchip. 2.7V 4-Channel/8-Channel 10-Bit A/D Converters with SPI Serial Inter- face . `https://cdn-shop.adafruit.com/datasheets/MCP3008.pdf/` (2008). [Online; accessed 10-June-2018].

[14] Povey, D. *et al.* The kaldi speech recognition toolkit (2011). URL `http://infoscience.epfl.ch/record/192584`. IEEE Catalog No.: CFP11SRW-USB.

[15] Luque, J., Segura, C., Sánchez, A., Umbert, M. & Galindo, L. The role of linguistic and prosodic cues on the prediction of self-reported satisfaction in contact centre phone calls. 2346–2350 (2017).