Bag of features for imagined speech classification in electroencephalograms

Por:

Jesús Salvador García Salinas

Tesis sometida como requisito parcial para obtener el grado de:

MAESTRO EN CIENCIAS EN EL ÁREA CIENCIAS COMPUTACIONALES

En el:

Instituto Nacional de Astrofísica, Óptica y Electrónica
Agosto, 2017
Tonantzintla, Puebla

Supervisada por:

Luis Villaseñor Pineda
Carlos Alberto Reyes García

Contents

List of Figures

List of Tables

Abstract

The interest for using of brain computer interfaces as a communication channel has been increasing nowadays, however, there are many challenges to achieve natural communication with this tool. On the particular case of imagined speech based brain computer interfaces, here still exists difficulties to extract information from the brain signals.

The objective of this work is to propose a representation based on characteristic units focused on EEG signals generated in imagined speech. From this characteristic units a representation will be developed, which, more than recognize a specific vocabulary, allows to extend such vocabulary.

In this work, a set of characteristic units or *bag of features* representation is explored. These type of representations have shown to be useful in similar tasks. Nevertheless, to determine an adequate bag of features for a specific problem requires to adjust many parameters.

The proposed method aims to find an automatic signal characterization obtaining *characteristic units* and later generating a representative pattern from them. This method finds a set of characteristic units from each class (i.e. each imagined word), which are used for recognition and classification of the imagined vocabulary of a subject. The generation of characteristic units is performed by a clustering method. The generated prototypes are considered the characteristic units, and are

called *codewords*. Each codeword is an instance from a general dictionary called *codebook*.

For evaluating the method, a database composed of the electroencephalograms from twenty seven Spanish native speakers was used. The data consists of five Spanish imagined words (”Arriba”, ”Abajo”, ”Izquierda”, ”Derecha”, ”Seleccionar”) repeated thirty three times each one, with a rest period between repetitions, this database was obtained in [Torres-García et al., 2013]. The proposed method achieved comparable results with related works, also have been tested in different approaches (i.e. transfer learning).

Bag of features is able to incorporate frequency, temporal and spatial information from the data. Also, different representations which consider information of all channels, and feature extraction methods were explored. In further steps, is expected that extracted characteristic units of the signals allow to use transfer learning to recognize new imagined words, these units can be seen as prototypes of each imagined word.

The calibration of brain computer interfaces is an important step, which requires abundant training data. Moreover, this calibration depends in the EEG device and its electrodes conductive properties. To address this issue, a transfer learning approach was explored to add new imagined words and to extend the model to new users.

Chapter 1

Introduction

There is a growing interest on Brain-Computer Interfaces (BCI). Initially this interest arose as a new communication channel to disabled persons. However, due to the reduction in cost of cerebral signal measuring devices, nowadays this new communication method is available to the public in general.

To control a device through a BCI, the user must produce a brain activity pattern. It can be evoked internally or by an external stimuli, and it will be identified by the system and transformed into commands for such device. The brain activity measurement can be taken by different devices. In this work electroencephalograms (EEG) are used to record brain electrophysiological signal. Moreover, this work is focused on brain signals evoked by imagined speech, i.e. to imagine a word diction without emitting nor articulating any sound.

Early approaches of BCIs were based on code generations to communicate [Dewan, 1967] (i.e. Morse code) and external stimuli response [Farwell and Donchin, 1988], where the brain activity was related to a stimuli that is well known. A different approach proposes the use of cognitive processes related to language like imagined speech, that is the action of imagining the diction of a word without emitting nor articulating any sound. The imagined speech is a conscious process which requires processing of the signal to be detected and used on BCIs, due this needs, different methods for imagined speech classification have been developed.

Despite the large amount of computational methods for processing, characterization and classification of brain signals on EEG [Lotte et al., 2007]; the extracted signals from imagined speech themselves owns properties which complicate their analysis (i.e. nonstationarity, nonlinearity and noise) [Klonowski, 2009]. Many solutions have been proposed to perform this task (see section 3.2), however, important challenges are still present to achieve a natural and fluid communication through an imagined speech based BCI.

The present work shows the generation of an imagined speech representation from the bag of features (BoF) method. Bag of features aims to an automatic signal characterization obtaining, at first instance, *characteristic units* and later generating a representative pattern from them. This method finds a set of characteristic units from each class (i.e. each imagined word), and these sets are used for recognition and classification of the imagined vocabulary of a subject. The generation of characteristic units is performed by a clustering method. The generated prototypes are considered the characteristic units, and are called *codewords*. Each codeword is an instance from a general dictionary called *codebook*.

1.1 Problem

In general biosignals have three properties, noise, nonlinearity and nonstationarity [Klonowski, 2009]. The noise could be inherent to the human body (i.e. produced by heart beating, blinking or respiration), or be produced by the recording devices. The nonlinearity of the signal makes difficult to adjust a model which predict its behavior. Nevertheless, to analyze segments of the signal as the Short Time Fourier

Transform does, allows to analyze the linearity on the signal. The nonstationarity is because the signal sources is in constant motion during the cognitive processes, also they have a variable duration over time.

In practice, the electric potentials based BCIs requires long training time to achieve the conscious control of brain signals, or could require an external stimuli to generate an specific brain response [Dewan, 1967, Farwell and Donchin, 1988]. To incorporate BCIs as a common use tool, is desirable that BCIs can identify the cognitive processes of the brain. The imagined speech is a conscious process, thus, it does not require previous training, but higher processing to obtain representative features from it.

For signals classification, many characterization techniques have been used, the most commons are based on frequency domain transforms (FFT, DWT) and have obtained diverse results using different processing methods and representations. Related works have encountered complications on multi-class imagined speech classification, multi-class classification of imagined speech has not achieved the same results as binary approaches. Also, some of the proposed representations have not explored the integration of temporal and spatial information of the EEG signal, these information may allow to detect patterns that previous representations did not found.

Due to a robust and simple representation is desirable, the proposed solution is based on a Bag of Features method which is able to incorporate frequency, temporal and spatial information from the data. An important step of the proposed method is the signal representation, the EEG signals are normally considered as a long vector in time where each channel represents a different signal, and different feature extraction methods are applied to them. Also, different representations which consider information of all channels, and feature extraction methods were explored. In further steps, is expected that extracted characteristic units of the signals allow to use transfer learning to recognize new imagined words, these units can be seen as prototypes of each imagined word, thus, if prototypes represent its classes well, they could discriminate among different classes.

Currently, the BCIs are hindered by time consuming calibrations before each use. The calibration of BCIs is an important step, which requires abundant training data, due to the brain signals noise and nonstationarity [Wang et al., 2015]. Moreover, this calibration depends in the EEG device and its electrodes conductive properties. To address this issue, a transfer learning approach was explored to add new imagined words and to extend the model to new users.

1.2 Research question and hypothesis

To address the problems presented for imagined speech classification the following research question is presented:

- How the bag of features could be adapted to represent imagined speech on EEG for classification improvement?

Then, the hypothesis is defined as:

- Bag of features representation can increase the patterns detection and improve the multi-class classification accuracy of imagined speech in comparison to the related work.

1.3 Objective

1.3.1 General Objective

To propose an imagined speech representation for its classification, through a bag of features method.

1.3.2 Specific Objectives

The specific objectives are listed below.

- Propose new representations for the EEG signals based on frequency, space and time features.
- Evaluate the proposed representations applied in a Bag of Features method for imagined speech classification.
- Apply transfer learning to the proposed method to extend the imagined speech vocabulary and to extend the model to new users.

1.4 Scope and limitations

This work is focus on the analysis of EEG signal of imagined speech task over available databases. There are few imagined speech public databases, and each one has its own acquisition protocol. Nevertheless the obtained results are compared with related work. The bag of features is extended with methods from different areas, using different processes to transform the data.

Chapter 2

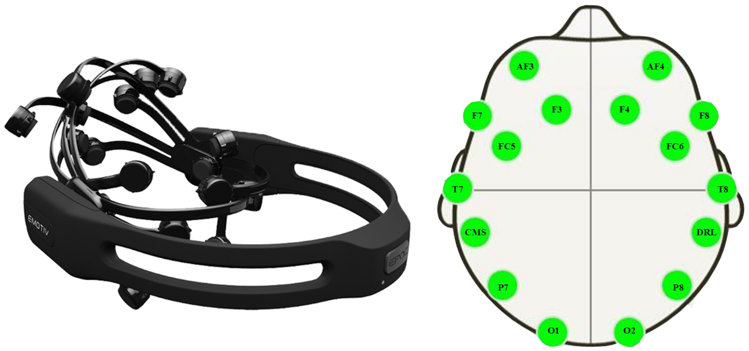
Theoretical Framework

This Chapter contains a brief explanation of the applied methods, devices and signals. Section 2.1 introduces a description of BCI classes and its operation due to BCIs are the devices in which is desired to implement the proposed method. In Section 2.2 the measure of the biosignal of interest (i.e. brain electrical current) by means of the EEG is explained. Later, Section 2.3 explains BoF method, this basic explanation will help to introduce the proposed representations and data processing. Section 2.4 objective is to explain the feature extraction methods applied in the signal for the BoF representation. Another BoF component is reviewed in Section 2.5, the clustering step, where different clustering methods are analyzed and compared. Finally, the Section 2.6 comprises the classification method which will be applied to the BoF results.

2.1 Brain Computer Interfaces

A BCI is a computer-based system that acquires brain signals, analyzes them, and translates them into commands that are relayed to an output device to carry out a desired action [Shih et al., 2012]. It can be seen as a communication system that does not require peripheral muscular activity. BCIs enable subjects to send commands to an electronic device only by means of neural activity [Lotte et al., 2007].

Normal human brain activity produces a wide variety of signals that can be measured and that have potential use in a BCI. These signals include electrical, magnetic, metabolic, chemical, thermal and mechanical responses to brain activity. These signals can be detected with appropriately designed sensors. Electrical currents produced by synchronized synaptic activities can be measured by (in order of invasiveness) scalp EEG (see fig. 2.1), epidural electrodes and electrocorticography (ECoG). Action potentials from neurons can be recorded using magnetoencephalographic (MEG) activity. Metabolic consequences of neural activity include changes in blood flow and metabolism, which can be imaged using functional magnetic resonance imaging (fMRI), positron emission tomography (PET), and optical techniques as functional near-infrared spectroscopy (fNIRS) [Vaughan, 2003].



(a) EPOC

(b) Electrodes position

Figure 2.1: EMOTIV EEG device [EMOTIV, 2017]

In [Wolpaw et al., 2002] the BCIs are divided in two classes, dependent and independent. Dependent BCIs do not use the brain's normal output pathways to carry a message, but activity in these pathways is needed to generate the brain activity that carries it. For example, one dependent BCI presents the user with a matrix of letters that flash one at time, and the user selects a specific letter by looking directly at it, in this case, the brain's output channel is EEG, but the generation of the EEG signal depends on the gaze direction, and therefore on extra-ocular muscles and cranial nerves that activate them.

Independent BCIs do not depend in any way on the brain's normal output pathways, the message is not carried by peripheral nerves and muscles. For example, one independent BCI presents the user with a matrix of letters that flash one at time, and the user selects a specific letter by producing a P300 evoked potential when that letter flashes. In this case, the generation of the signal depends mainly on the user intent not on the orientation of the eyes.

Also, a classification of present day BCIs is made, creating five groups. The first group, visual evoked potentials, are dependent BCIs, the other four groups, those using slow cortical potentials, P300 evoked potentials, mu and beta rhythms, and cortical neuronal action, are believed to be independent BCIs.

- Visual evoked potentials

  The Visual Evoked Potentials (VEP) - based communication systems depend on the user's ability to control gaze direction, these systems have the same function as gaze direction determining systems. This potentials are generated by a visual stimuli and are taken from visual cortex. Actually this systems determine the direction using a set of flashes with different frequency rates, and search for a match of the signal and a flash frequency to identify the user selection.

- Slow cortical potentials

  Among the lowest frequency features of the EEG, are slow voltage changes generated in cortex. These potential shifts occur over 0.5 - 10 seconds and are called slow cortical potentials (SCPs). Negative SCPs are typically associated with movement and other functions involving cortical activation, while positive SCPs are usually associated with reduced cortical activation. This potentials can be controlled with previous training and can be used to control the movement of an object on a computer.

- P300 evoked potentials

  Infrequent or particularly significant auditory, visual or somatosensory stimuli; when interspersed with frequent or routine stimuli, typically evoke in the EEG over parietal cortex a positive peak at about 300 milliseconds. The advantage in this model is that it requires no training, but in a long term, the P300 signals habituate to the BCI tasks. The habituation of signals affects the BCIs, which require to adapt for changes in the signal.

- Mu and beta rhythms

  In awake people, primary sensory or motor cortical areas often display 8- 12 Hz activity when they are in an idle state, this is called mu rhythm when is focused over somatosensory or motor cortex, and the visual alpha rhythm when is focused over visual cortex. The mu rhythms are usually associated to 18 - 26 Hz beta rhythms. Some of these beta rhythms are harmonics of mu, but some others are separable rhythms. The movement or preparation for movement is typically associated with a decrease in mu and beta rhythms, and oppositely the rhythms increase after movement and with relaxation.

- Cortical neurons

  The use of microelectrodes to record single neurons in the cerebral cortices leads to a different form of BCI. This model requires that the user learns to control the discharge of single neurons in the motor cortex [Wang et al., 2013].

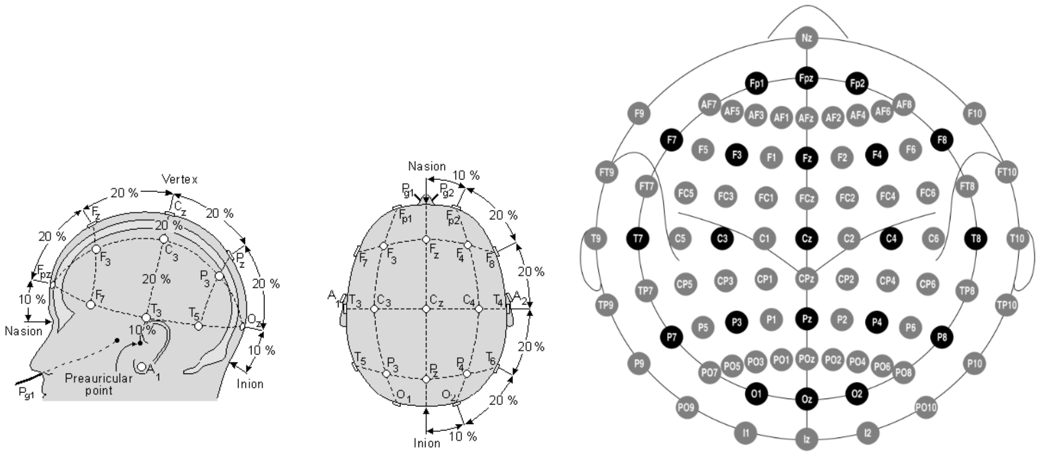
2.2 Electroencephalogram (EEG)

Our brain generates electrical currents with variable voltage, the voltage variation represents cerebral activity and can be measured by an EEG in a scale of microvolts [Mohamed and Justin, 2013].

The EEG consists of the summed electrical activities of populations of neurons, with a modest contribution from glial cells. The neurons are excitable cells with characteristic intrinsic electrical properties, and their activities produce electrical and magnetic fields. These fields may be recorded by means of electrodes at short distance (e.g. local field potentials), from cortical surface (e.g. ECoG), or at longer distances as the scalp (e.g. EEG) [da Silva, 2009].

The EEG registers rhythmic cerebral activity in many channels that represent the area of measurement. The number of channels may vary and a number of channels cannot be established due to the digital acquisition methods. Nevertheless, on daily clinical practice the equipments are provided among 8, 16, 32, 64 or 128 channels [Zarate, 2005]. The neuronal activity is obtained by electrodes over the scalp and it is amplified by electronic devices. The EEG spatial resolution is poor, but the temporal resolution is excellent, currently the main activity in processing EEG is signal analysis and pattern recognition of the temporal information of the signal [Toennies, 2012].

The electrodes positioning is commonly defined by the 10-20 standard system, shown in fig. 2.2, which is based in the relation among the electrodes position and the cerebral cortex area. The numbers 10 and 20 are the distances between adjacent electrodes, and represent 10% and 20% of the total distance of the skull from the front to the back side, or from left to right side. The reference points for this measures are the nasion, located between the nose and the forehead; the inion, which is the lower part of the skull; the vertex, which is the center point on the top


Vertex
20 %
C_z
F_z
20 %
F_3
C_3
P_3
P_z
F_pz
10 %
Nasion
F_7
T_3
T_5
O_z
10 %
Preauricular point
A_1
P_g1
Inion
Nasion
10 %
P_g1
P_g2
F_p1
F_p2
F_7
F_3
F_z
F_4
F_8
A_1
T_3
C_3
C_z
C_4
T_4
A_2
T_5
P_3
P_z
P_4
T_6
O_1
O_2
Inion
Nz
Fp1 Fpz Fp2
AF7 AF5 AF3 AF1 AFz AF2 AF4 AF6 AF8
F9 F7 F5 F3 F1 Fz F2 F4 F6 F8 F10
FT9 FT7 FC5 FC3 FC1 FCz FC2 FC4 FC6 FT8 FT10
T9 T7 C5 C3 C1 Cz C2 C4 C6 T8 T10
TP9 TP7 CP5 CP3 CP1 CPz CP2 CP4 CP6 TP8 TP10
P9 P7 P5 P3 P1 Pz P2 P4 P6 P8 P10
PO9 PO7 PO5 PO3 PO1 POz PO2 PO4 PO6 PO8 PO10
O1 Oz O2
I1 Iz I2

(a) 10-20 system measurement [Simkin et al., 2014]

(b) 10-20 system positions [Oostenveld and Praamstra, 2001]

Figure 2.2: Standardized position by the Electroencephalographic American Society. Black dots indicate the original positions, gray dots are introduced in the 10-10 extension

of the skull; and the pre-auricular points anterior to the ears [Technologies, 2012].

Each electrode has a letter and a number to identify its precise position. Electrodes are divided in frontal, central, temporal, parietal and occipital lobes, and are recognized by the letters F, C, T, P and O, respectively. The subindex indicates the hemisphere and position of the electrode, the even numbers represent the right hemisphere and the odd numbers the left hemisphere [Zarate, 2005].

The EEG analysis is usually described in terms of rhythmic activity, which is commonly divided in specific frequency bands, 0.5-4 Hz(delta), 4-8 Hz (theta), 8-10 Hz (low alpha), 10-12 Hz (high alpha), 12-30 Hz (beta) and 30-100 Hz (gamma).

2.3 Bag of features (BoF)

This method is based on the *Vector Quantization* traditional approach, which objective is to achieve an automatic signal characterization, discretizing its representation. In signal analysis area, many variations and adaptations of the method have been developed; and receives different names according to the application area, is referred to as a bag of words in document classification, bag of instances in multiple instance learning, bag of frames in audio and speech recognition, bag of patterns in signal processing and pattern recognition and bag of images or visual words in computer vision [Baydogan et al., 2013].

The bag of features model was originally developed for document representation. Its basic idea is to define a codebook that contains a set of codewords and then represent a document, a signal or an image, as a histogram of the generated codewords. Although the order information of words is ignored, the bag of words model is very effective to capture document information [Wang et al., 2013]. The codewords are taken from the codebook, which, in signal analysis, is commonly generated by a clustering procedure (e.g. k-means) over segments of the analyzed object, later each extracted segment is associated to a codeword, then it is possible to generate histograms to analyze.

A formal definition of bag o words representation is reviewed as follows. A time series is defined by the vector $x^i = (x^i_1, x^i_2, ..., x^i_p)$, for p samples. Each instance x^i is associated with a class y^i for $i = 1, 2, ..., n$ and $y^i \in \{1, 2, ..., C\}$ where n is the instance number, and C is the classes number. To extract local patterns, a sliding window w over the time series is needed. The movement m of the window cannot be greater than the window size $m \leq w$. The extracted subsequences will be $\lceil \frac{p-w+1}{m} \rceil$, thus, the data set will have $n(\lceil \frac{p-w+1}{m} \rceil)$ subsequences. Later, a clustering method is applied with k centroids, that will become the *codewords* of the *codebook*, $K \in \mathbb{R}^{(w \times d)}$

[Gui and Yeh, 2014].

In summary, in signal processing approach, the signal is segmented and representative units are generated by means of a clustering technique, these clusters prototypes receive the name of *codewords* and together are named as *codebook*. Once the codebook is created, the segments of the signal are taken and for each one the most similar codeword is assigned to it. Next step is to create a histogram of the present codewords on the signal.

This technique assumes that a set of objects consist of finite features and that objects are unordered sets of features occurrences, also, histogram based approaches ignore the temporal ordering and therefore may not identify a specific content or pattern.

2.4 Feature extraction

2.4.1 Fast Fourier Transform

The Discrete Fourier Transform (DFT) is a discrete transform which obtains a frequency domain representation from a function given in time frequency.

The DFT of an array X of n complex numbers is the array ¶ given by Eq. 2.1,

$$\Phi[j] = \sum_{i=0}^{n-1} X[i]\omega_n^{ij} \tag{2.1}$$

where $0 \leq j \leq n$ and $\omega_n = \exp(-2\pi\sqrt{-1}/n)$. This function requires $\Theta(n^2)$ operations. Fast Fourier Transform (FFT) are $O(n \log n)$ algorithms to compute approximately the same result as DFT [Frigo and Johnson, 2005].

The most used algorithm is known as Cooley-Tukey [Cooley and Tukey, 1965], who rediscovered it from a Gauss work [Heideman et al., 1984]. This algorithm aims to transform a DFT of size $n = n_1n_2$ to smaller DFTs of sizes $n_1 \times n_2$. Letting $i = i_1n_2 + i_2$ and $j = j_1 + j2n1$, we have Eq. 2.2 from Eq. 2.1.

$$\Phi[j_1 + j_2n_1] = \sum_{i_2=0}^{n_2-1}[(\sum_{i_1}^{n_1-1} X[i_1n_2 + j_2]\omega_{n_1}^{i_1j_1})\omega_n^{i_2j_1}]\omega_{n_2^{i_2j_2}} \tag{2.2}$$

The algorithm computes n_2 DFTs of size n_1 (inner sum), multiplies the result by the factors $\omega_n^{i_2j_1}$, and finally computes n_1 DFTs of size n_2 (outer sum), this decomposition is then continued recursively.

In summary, the algorithm divides the data points in tow groups, even and odds, recursively until there are groups of two points. These groups are operated in crossed operations called *butterfly* (see fig. 2.3), each operation returns the DFT multiplying both points by factors $\omega_n^{i_2j_1}$, and then adding or subtracting the results

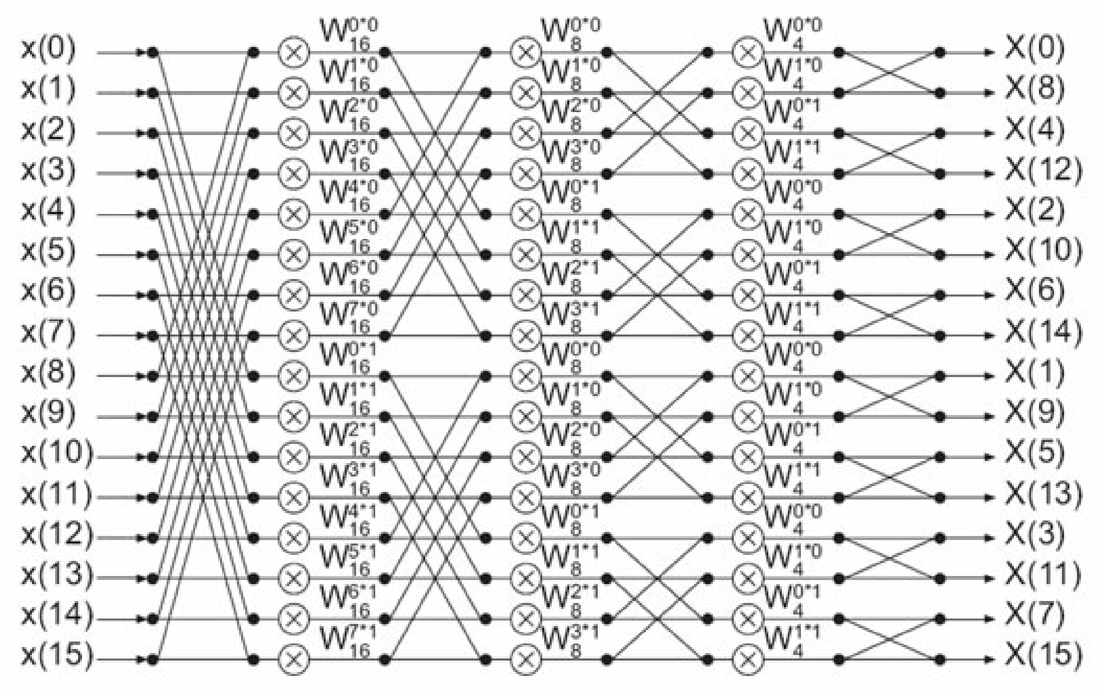


Figure 2.3: FFT butterfly example with 16 points [CDS, 2017]

2.4.2 Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) in comparison with DFT, replaces the infinitely oscillating sinusoidal basis functions with a set of locally oscillating basis functions called wavelets. This wavelets are stretched and shifted versions of a fundamental, real-valued band-pass wavelet $\psi(x)$. In combination with a low-pass scaling function $\rho(x)$, they form an orthonormal basis expansion for one dimensional real valued continuous time signals [Selesnick et al., 2005].

The DWT is defined as,

$$\Upsilon\rho(j_0, \xi) = \frac{1}{\sqrt{n}} \sum_{x} f(x)\rho_{j_0,\xi}(x) \tag{2.3}$$

$$\Upsilon\psi(j, \xi) = \frac{1}{\sqrt{n}} \sum_{k} f(x)\psi_{j,\xi}(x) \tag{2.4}$$

for $j \geq j_0$, usually $j_0 = 0$ and n is a power of 2. Where $f(x)$, $\rho(x)$ and $\psi(x)$ are functions of a discrete variable $x = 0, 1, 2, ..., n-1$, and $\xi = 0, 1, 2..., 2^j$.

The coefficients defined in Eq. 2.3 and 2.4 are called approximation and detail coefficients, respectively [Martinez and Escribano, 2008]. This coefficients can be seen in Figure 2.4, where s is a signal of length N

The function $\rho_{j,\xi}(x)$ is a member of the set of expansion functions derived from a scaling function $\rho(x)$, by translation and scaling using,

$$\rho_{j,\xi}(x) = 2^{j/2}\rho(2^j x - \xi) \tag{2.5}$$

The function $\psi_{j,\xi}$ is a member of the set of wavelets derived from a wavelet

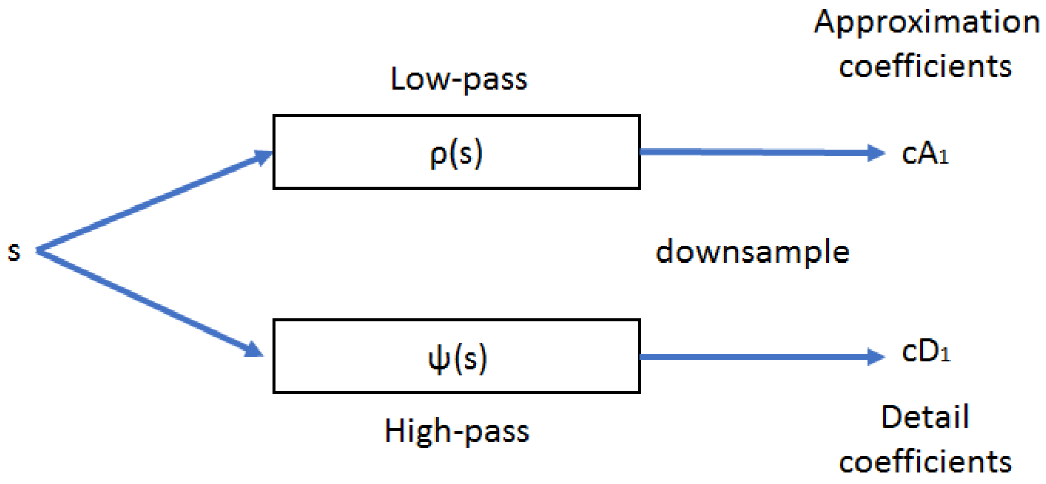


Figure 2.4: Wavelet decomposition [MathWorks, 2017]

function $\psi(x)$, by translation and scaling using,

$$\psi_{j,\xi}(x) = 2^{j/2}\psi(2^j x - \xi) \tag{2.6}$$

Then the DWT can be formulated as a filtering operation with two filters, low-pass filter h_ρ and high-pass filter h_ψ. As seen in fig. 2.5, given a signal s of length N, the DWT consists of $\log_2 N$ stages at most. The first step produces two sets of coefficients, this operations can be repeated as a cascade on the the approximation coefficients.

2.5 Clustering

2.5.1 k-Means++

Lloyd's algorithm, known as k-means is a common clustering technique that aims to minimize the distance between a set of points in a defined cluster. The k-means

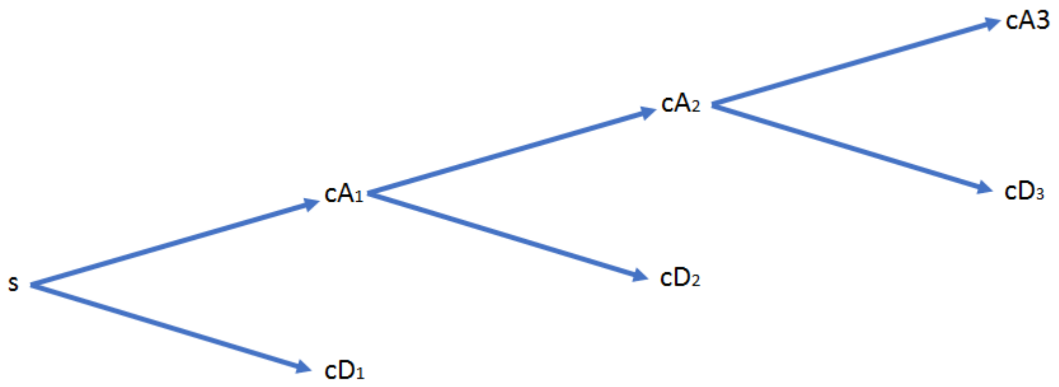


Figure 2.5: Wavelet levels [MathWorks, 2017]

requires an integer value k and a set of n data points $X \subset \mathbb{R}^d$. The objective is to choose k centers C that minimize the Eq. 2.7.

$$\phi = \sum_{x \in X} \min_{k \in K} ||x - k||^2 \tag{2.7}$$

The standard k-means algorithm which computes the Eq. 2.7, is simple and generally fast for small datasets. Nevertheless, it has an NP-hard complexity. If the clusters number ($|K|$) and the dimensional space (d) are fixed, the problem can be exactly solved in $O(n^{d|K|+1})$, where n is the number of entities to be clustered. The algorithm begins with $|K|$ arbitrary centers, typically chosen randomly from the data points. Each point is assigned to the nearest center and then, each center is recomputed to be the center of mass of the points assigned to it.

A different way of cluster initialization, proposed in [Arthur and Vassilvitskii, 2007], improves the standard k-means, and its named k-means++.

Let $D(X)$ denote the shortest distance from a data point x to the closest center, then we have.

The result is a $O(\log k)$ algorithm that overcomes the standard k-means implementation. Among the disadvantages of the method, the number of clusters must

Algorithm 1 k-Means

```
1: Arbitrarily choose n initial centers $K = k_1, ..., k_n$.
2: For each $l \in 1, ..., n$, set the cluster $k_l$ to be the set of points in $x$ that are closer to $k_l$ than they are to $k_{l'}$ for all $l' \neq l$.
3: For each $l \in 1, ..., n$, set $k_l$ to be the center of mass of all points in $K_l : k_l = \frac{1}{|k_l|} \sum_{k' \in K_l} k'$.
4: Repeat Steps 2 and 3 until $K$ no longer changes.
```

Algorithm 2 k-Means++

```
1: Choose an initial center $k_l$ uniformly at random from $X$.
2: Choose the next center $k_l$, selecting $k_l = x' \in X$ with probability $\frac{D(x')^2}{\sum_{x\ inX} D(x)^2}$
3: Repeat Step 1 until we have chosen a total of $n$ centers
4: Proceed as with the standard k-means algorithm
```

be established *a priori*, to select an optimal value, heuristic information or an incremental iteration of the method is needed. An example result is shown in fig. 2.6.

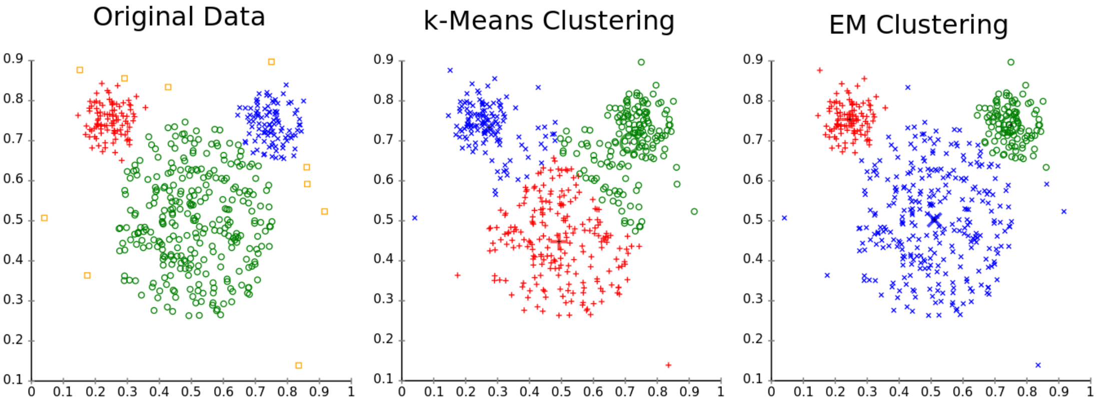


Figure 2.6: K-means and EM clustering of a data set [Wikipedia, 2017]

2.5.2 Expectation Maximization

The Expectation Maximization (EM), is an iterative method to estimate unknown parameters Θ, given measurement data U [Dellaert, 2002]. The objective is to maximize the posterior probability of Θ, marginalizing a set of unknown variables E,

$$\Theta^* = argmax_{\Theta} \sum_{E \in \Sigma^n} P(\Theta, E|U) \tag{2.8}$$

The process alternates between estimating the unknowns Θ and the hidden variables J. However, instead of finding the best $E \in \Sigma$, EM computes a distribution over the space Σ.

In summary, the EM algorithm is an iterative algorithm which alternates between two steps, E-step and M-step. For clustering, EM makes use of Gaussian Mixture Models (GMM) and estimates a set of parameters until a convergence is achieved, with the following algorithm [Jin and Han, 2010].

Algorithm 3 EM clustering

```
Set initial parameters $\mu$ and $\sigma$ (mean and deviation respectively for normal distribution) randomly.
Iteratively refine parameters with E and M steps until convergence is reached
E-step: Compute the membership probabilities for each instance based on initial parameters.
M-step: Recompute the parameters based on the new membership probabilities.
Assign each instance to the cluster with which it has the highest membership probabilities.
```

The method convergence is usually set on the instances probabilities changing, when this change is lower than a fixed ϵ, the algorithm stops.

In comparison with k-Means which makes an spherical clustering, this method allows a better cluster adaptation to the data due to the cluster assign is not based on distances. Moreover, EM algorithm establish an optimal clusters number considering the convergence threshold. The disadvantage compared to k-means is the slow convergence of the algorithm

2.6 Classification

2.6.1 Multinomial Naive Bayes

Thinking in Multinomial Naive Bayes for text classification, a document is treated as a sequence of words and it is assumed that each word position is generated independently of every other. For classification, its assume that there is a number of classes $c \in \{1, 2, ..., n\}$, the parameter vector for a class is given by $\theta_z = \{\tau z1, \tau_{z2}, ..., \tau_{zn}\}$, where n is the size of the vocabulary, $\sum_i \tau_{ci} = 1$ and τ_{zi} is the probability that words i occurs in that class. The likelihood of a document is a product of the parameters of the words that appear in the document,

$$\Xi(\mu|\tau z) = \frac{(\Sigma_i f_i)!}{\Pi_i f_i!} \Pi_i (\tau_{zi})^{f_i} \tag{2.9}$$

where f_i is the frequency count of word i in document μ. By assigning a prior distribution over the set of classes, $\Xi(\tau_c)$, we can arrive at the minimum error classification rule which selects the class with the largest posterior probability [Rennie et al., 2003].

In summary, with a multinomial event model, samples represent the frequencies in which certain events have been generated by a multinomial $(s_1, ..., s_n)$ where s_i is the probability that event i occurs. A feature vector $t = (t_1, ..., t_n)$ is then a histogram.

Chapter 3

Related work

In this Chapter, the previous work in imagined speech and BoF will be reviewed. Section 3.1 begins with an introduction to the first BCI approaches. Later, Section 3.2 compiles imagined speech classification works and them different approaches. In Section 3.3 some works related to BoF method are reviewed. And finally, Section 3.4 shows a discussion of the related work and the proposed method.

3.2

3.1 First approaches

Communication by means of brain signals started with the detection of electric potentials indirectly related to the speech cognitive process, despite its disadvantages this scheme is used by some BCIs.

One of the first reported works, presented on [Dewan, 1967], made use of activation and blocking of alpha rhythms, a frequency range of brain signals, to generate Morse code. This requires a previous training that consists on manipulating the oculomotor configuration to achieve the alpha rhythm control.

In [Farwell and Donchin, 1988] P300 signals were used to detect visual responses on characters, through an alphabet shown in a screen. This system detects the brain response to a visual stimuli that moves along the alphabet characters, the brain response indicate then which character the user tried to communicate.

3.2 Imagined speech

First approaches on BCIs imply that users can generate specific cerebral signals, or take advantage of natural cerebral responses to external stimuli. An imagined speech based BCI makes use of signals generated by thee cognitive process of speech. The advantage is that the user doesn't need a training to generate specific cerebral signals.

Different works related to imagined speech are shown below. They differ not only on the proposed method but on the evaluation, due to the experimental designs use different subjects, acquisition protocols and imagined speech vocabularies.

[DaSalla et al., 2009] proposes the classification of two imagined vowels through common spatial patterns (CSP), support vector machines (SVM) and band pass filters, achieving an accuracy of $62.6 \pm 8.3\%$ for three subjects.

Classification of imagined words began with [Suppes et al., 1997], which analyzes EEG and MEG signals to classify among seven words (first, second, third, yes, no, right,left). Characterization is based on Fast Fourier Transform (FFT) and a band pass filter, to apply later an Inverse Fast Fourier Transform (IFFT). The signals were compared by means of least squares with a prototype created from the mean of themselves, obtaining an accuracy of $52.57 \pm 20\%$ for five subjects.

In [Torres-García et al., 2012] a Spanish vocabulary of five Spanish words (”Arriba”, ”Abajo”, ”Izquierda”, ”Derecha”, ”Seleccionar”) is proposed. Channels near the language area and a band pass filtering between 4 and 25 Hz were used. Characterization with Discrete Wavelet Transform (DWT) was applied to train four classifiers: Naive Bayes, Random Forest, Support Vector Machines and Bagging Random Forest. Best results were obtained by Bagging Random Forest with an accuracy of $41.96 \pm 3\%$ for three subjects.

A simpler scheme is shown in [Salama et al., 2014], where two Arabic words (Yes, No) are classified. Low alpha, high alpha, low beta and high beta rhythms of one channel EEG were analyzed with two methods, the first one obtained statistical data of the signal (minimum, maximum and mean) and the second applied DWT with six decomposition levels. The classification was performed by Support Vector Machines, Linear Discriminant Analysis, Self Organized Maps, Multilayer Perceptron and assemblies of them, mean accuracy obtained was 56% for a set of seven subjects.

Better classification results were obtained in [Kim et al., 2013], where eight words were classified with two different semantic classes related to the human face and numbers, using 30 channels and two subjects. The improvement implemented was a spatial-temporal pattern search, that obtained a mean accuracy of 92.46% with a Support Vector Machine.

A different approach was presented in [Zhao and Rudzicz, 2015], where a set of words was labeled according its phonemic and phonological features, the experimental set includes EEG, face and audio recordings. Many schemes of binary classification were explored like vowel vs consonant, presence of nasal, presence of bilabial, presence of high-front vowel and presence of high back vowel, the best recognition rate using only EEG recordings was 63.5% and obtained by presence of nasal with an SVM-quad classifier. This data includes sixty four channels from twelve subjects.

Recently, in [Torres-García et al., 2016] different wavelet families and classifiers were explored for multi-class imagined speech classification, the data set is formed by five Spanish words (”Arriba”, ”Abajo”, ”Izquierda”, ”Derecha”, ”Seleccionar”) and twenty seven subjects. Also an automatic channel selection based on fuzzy inference were implemented to reduce the data set, and an accuracy of $68.18 \pm 16\%$ were achieved.

In [Pressel Coretto et al., 2017], a larger vocabulary is presented, it includes the Spanish words (”Arriba”, ”Abajo”, ”Izquierda”, ”Derecha”, ”Adelante”, ”Atras”) which were recorded from fifteen subjects, and applies the method presented on [Torres-García et al., 2013], the obtained accuracy rate was $18.58 \pm 1.47\%$.

Table 3.1 shows a comparison of imagined speech related work and the present work, is important to remark that classes have different approaches in some works. Also the acquisition protocols change in every work.

3.3 Bag of features

BoF has been used in different areas of application. Moreover, as it will be seen in this Section, the BoF method has different improvements and modifications in every step.

In [Lin and Li, 2009], the bag of patterns representation is used to classify electrocardiogram (EKG) time series using hierarchical and partitional clustering, to create bag of patterns representations. As it may be seen, the obtained features, pre-processing and classification are all independents from the bag of patterns.

In the specific case of EEG signals, the bag of features was used by [Wang et al., 2013], where EEG (and EKG) signals were analyzed for epilepsy detection. One channel EEG signals were used, also features were extracted by DWT, this features were clustered by k-means algorithm. Histograms were created through 1-Nearest Neighbor and finally classified by 1-Nearest Neighbor too. The accuracy obtained was $87.8 \pm 2.3\%$.

There are works that implement modifications over the classic bag of features. [Ordonez et al., 2011] presents different ways of data representation for bag of patterns. The signals were preprocessed by a technique named Symbolic Aggregate approXimation (SAX) presented on [Lin et al., 2007] which converts the signal in a sequence of text. The first BoP method is called Multivariate Bag of Patterns, that captures the relationship between time series over the time, the method creates multi-variated words that represent the time series in one time interval. Another method is called Stacked Bag of Patterns where, unlike the later method, each signal is treated as an individual BoP instance and is later concatenated. The last method uses Adapted Natural Language Processing Techniques on the two previous methods, document processing techniques are applied on the SAX converted signal, Term Frequency (TF),Inverse Document Frequency (IDF), Inverse Frequency (IF) and the combination IDF-IF, to analyze time series of different lengths.

In [Plinge et al., 2014], a modified version of clustering is implemented and is called *Bag of Super-Features*. The method consists on generate clusters for each class on the data set and later join them. Because disregarding the labels in clustering step can lead to mitigation of significant differences [Lazebnik and Raginsky, 2009].

In an attempt to classify long term information, [Yeh et al., 2013] proposes a Dual Layer Bag of Frames Model (DLBoF), applied on music genre classification. This method applies a Bag of Frames with FFT characterization over a set of signals, the next step is to create the second layer BoF, this is achieved applying a Bag of Histograms Aggregation. Then a first layer dictionary is trained from FFT spectrograms and a second layer trained from histograms.

Using the BoP, signal temporal information is lost since the sequences of the extracted features is ignored to create a histogram representation.[Gui and Yeh, 2014] proposes a temporal BoW that consists on dividing a segmented signal and applying the BoW to each interval generated, obtaining many histograms from one signal segment. Then this histograms must be combined to create a new instance.

3.4 Discussion

Present work collects methods applied in [Ordonez et al., 2011, Wang et al., 2013, Plinge et al., 2014], which use different approaches of BoF. Nevertheless, as it can be seen in Table 3.2, the present work is able to implement temporal and spatial information, which, in combination with feature extraction step includes frequency information of the signal. It is expected that the inclusion of these features, representative patterns of imagined speech can be found.

The bio-signals related works for BoF apply an specific pre-processing method called SAX [Lin et al., 2007], which is a discretization method. Present work pre-processing method is simpler, in fact, the pre-processing methods could be applied in data acquisition step using physical filters if necessary, this could help to improve the practical applications of BCIs in real time.

Most of the BoF related works have been applied in a single channel signals recording. Moreover, these methods are not applied in bio-signals, present work attempts to include all the EEG channels for patterns search. Nevertheless, in practice the acquisition channel reduction is desirable, [Torres-García et al., 2016] address this task reducing the acquisition channels and improving imagined speech classification.

Imagined speech related work, see Table 3.1, commonly classifies among syllables, semantic groups or vowels, there are few reported works in which single words are classified [Zhao and Rudzicz, 2015, Torres-García et al., 2016, Pressel Coretto et al., 2017]. Besides, there is not a standard acquisition protocol for imagined speech EEG, thus, each imagined speech database is different. In this work different databases were analyzed with the proposed method. Nevertheless to compare the results with related work was not always possible.

An expected result classifying imagined speech with BoF, is that the generated codebooks will be able to generalize the imagined speech features. In consequence this will allow to extend the imagined speech vocabulary for each subject, and also to extend a generated codebook to different subjects.

Table 3.1: Imagined speech works comparison

Work	Method	Subjects	Classes	Electrodes	Classification
[DaSalla et al., 2009]	Common Spatial Patterns	3	2 vowels	64	SVM
[Kim et al., 2013]	Spatial-temporal pattern search	2	2 semantic groups	30	SVM Feature selection
[Salama et al., 2014]	Raw signal	7	2 syllables	1	Self Organizing Map
[Zhao and Rudzicz, 2015]	Statistical features	15	4 words	64	SVM
[Torres-García et al., 2016]	DWT Fuzzy inference	27	5 words	6.87	Random forest
[Pressel Coretto et al., 2017]	DWT	3	6 words	6	Random forest
Present work	Bag of Features	27	5 words	14	Multinomial Naive Bayes

Table 3.2: Bag of Words comparison

Work	Task	Pre processing	Clustering	Feature Extraction	Temporal Spatial
[Lin and Li, 2009] Temporal Bag of Patterns	EKG anomaly detection	SAX	Hierarchical Partitional	No	Yes / No
[Ordonez et al., 2011] Multivariate Bag of Patterns	Hypotensive episode detection	SAX	Hierarchical	No	No / Yes
[Ordonez et al., 2011] Stacked Bag of Patterns	Hypotensive episode detection	SAX	Hierarchical	No	No / Yes
[Wang et al., 2013] Bag of Features	Epilepsy detection	SAX	k-Means	DWT	No / No
[Yeh et al., 2013] Dual Layer Bag of Frames	Music genre classify	No	Sparce coding	Spectrogram extraction	No / No
[Gui and Yeh, 2014] Bag of Words	Door trajectory	No	k-Means	No	Yes / No
[Plinge et al., 2014] Bag of Super Features	Acoustic event detection	Gaussian Mixture Models	EM	GFCC MFCC	Yes / No
Present work Bag of Features	Imagined speech classify	Filtering CAR	k-Means EM	FFT DWT No	Yes / Yes

Chapter 4

Proposed method

4.1 General description

The general method flowchart is shown in Fig. 4.1, the first step is to extract features from the data through a predefined window that moves along the signal. The extracted features can be disposed in different orders as will be seen in Chapter 5. Later, a clustering method is applied to obtain the *codebook* from these features. This clustering is applied to the features from each class [Lazebnik and Raginsky, 2009] which are later concatenated in a complete codebook [Plinge et al., 2014]. Each prototype obtained from the clustering will receive the name of *codeword.*

The codebook was used to assign to each instance (i.e. extracted features windows) the value of a codeword, this transforms the signal in a sequence of previously known codewords, the next step is to generate a set of histograms from this sequences. Each histogram will be labeled according to its class (i.e. imagined word) resulting a set of instances for a classifying step.

The method is applied individually among subjects. The subject's data was split in 75% for training and 25% for testing. Each step select the data randomly for all repetitions and later the results are averaged to obtain a total result for each subject.

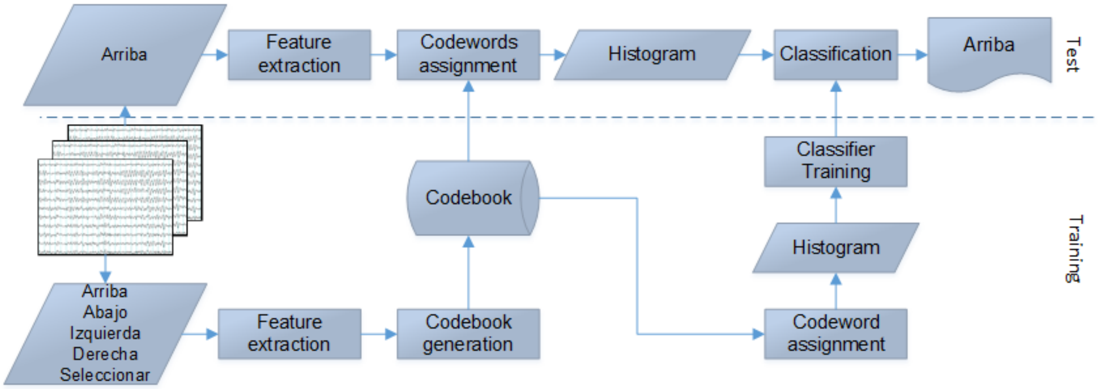


Figure 4.1: General method flowchart

The general method depends directly on the signal characterization for codewords generation. Thus, different representations were analyzed, combining frequency, temporal and spatial information of the signal. In a first approach frequency related features will be used by the idea that the different brain signal frequencies represent specific cognitive states [Wolpaw et al., 2002]. Furthermore, a spatial representation is explored, following the representations in [Ordonez et al., 2011], in which no signal processing is applied, the microvolt signal values are used as features taking into account all channels. And finally, a time representation is considered in combination with previous representations, it considers sequences of the instances obtained in this step and will be explained in section 4.5.

After introducing the detailed method description on section 4.3 and subsequent sections , the next section will describe the database, the acquisition protocol and the preprocessing of the signals.

4.2 Data

The imagined speech data set was built by [Torres-García et al., 2013]. The EEG of twenty seven native Spanish speaker subjects were registered through the EPOC kit of Emotiv, which has fourteen recording channels (i.e. AF3, AF4, F3, F4, F7, F8, FC5, FC6, P7, P8, T7, T8, O1, O2) and a 128 Hz sample rate. The data consists of five imagined speech Spanish words ("Arriba", "Abajo", "Izquierda", "Derecha", "Seleccionar") repeated thirty three times each one, with a rest period between repetitions. The acquisition protocol consisted in place a subject in a comfortable sit position in front of a monitor, which indicated the word to be imagined. The user must use a mouse to indicate the start and end of its imagined speech. This recordings were taken in a controlled environment without sonic nor visual noise. However, the acquisition protocol have two drawbacks, the users indication of the imagined speech start and end may introduce noise to the recorded signal. Moreover, the words were present to the user in sequence, to reduce bias a random selection of the words would be preferable.

The EPOC kit is a device with a fixed mounting scheme which electrical references are placed in P3 and P4, near the recording electrodes position, and are able to detect common cognitive and noise data. The data were processed with a Common Average Reference (CAR) (Eq. 4.1), which is able to delete the common information on all the channels in every sample [Ludwig et al., 2009, Alhaddad, 2012].

$$V_i^{CAR} = V_i^{ER} - \frac{1}{n}\sum_{j=1}^{n} V_j^{ER} \tag{4.1}$$

where V_i^{ER} is the electrical potential among the i^{th} electrode and the reference, and n is the number of electrodes.

In addition to reduce the noise, this method was useful to normalize data among subjects due to the mean subtraction. When mean is subtracted, every signal is reduced to a zero mean scale.

Also, a low-pass filter to reduce the noise was applied, such filter (see Fig. 4.2) is an infinite impulse response Butterworth filter with a stop-band frequency of 50 Hz and a pass-band frequency of 40 Hz.

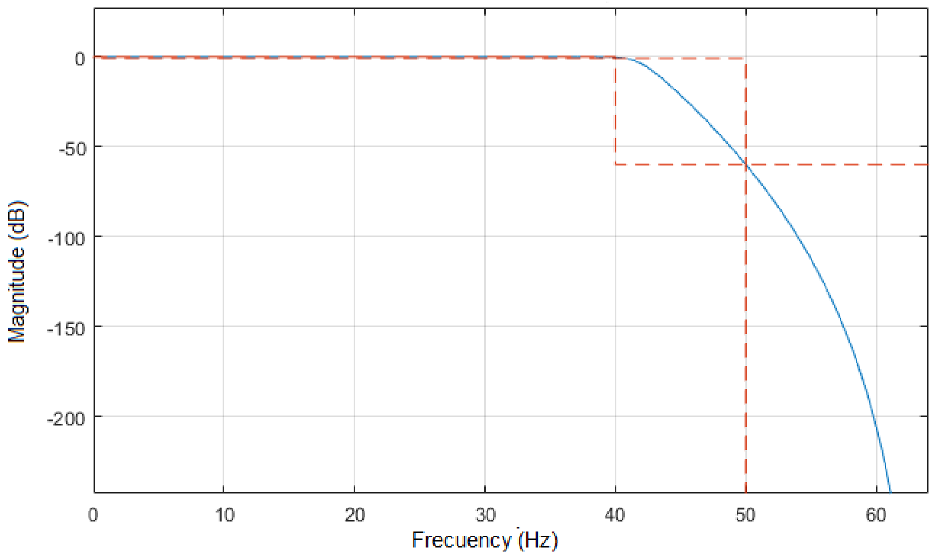


Figure 4.2: Low pass filter

4.3 Feature extraction

The feature extraction is realized by a sliding window through the signal and the extracted features can be ordered in different ways as will be seen in section 5. The standard method slides a window over one channel signal, and each window represents an instance for a clustering method in the next step, Fig. 4.3 shows an example over a signal which was CAR processed and filtered.

The feature extraction step allows to obtain temporal or spatial information of the signals. The ordering of the features will be analyzed in Chapter 5, standard

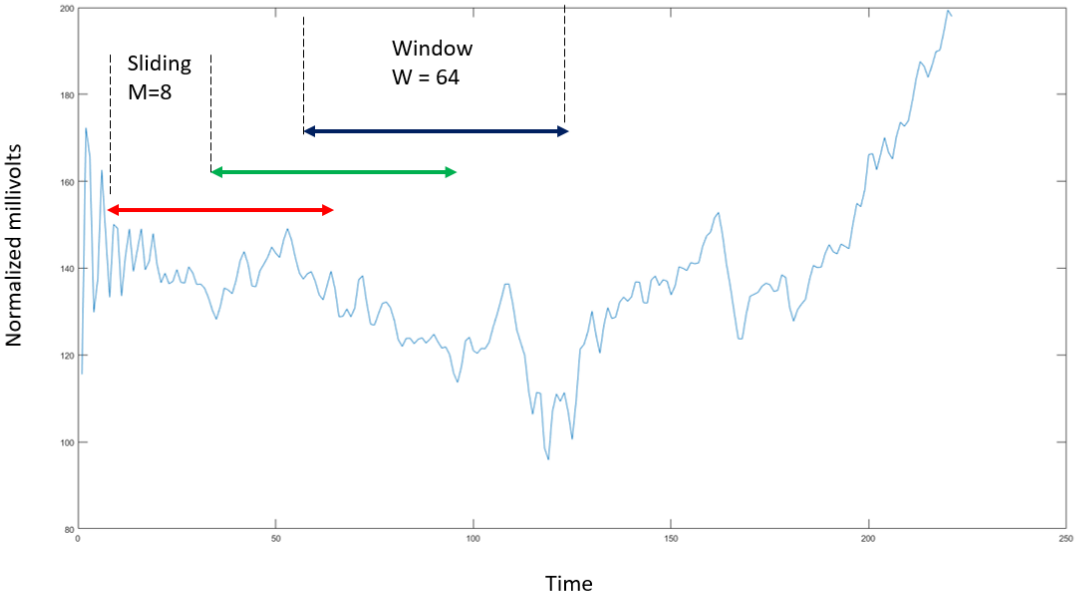


Figure 4.3: Feature extraction windows, each window step is shown in different colors.

method takes the extracted windows as independent instances of the signals and proceed to the clustering step.

The extracted features are the FFT absolute coefficients, the windows can be overlapped over time and can be seen as a Short Time Fourier Transform (STFT) implementation. Moreover, the windows are able to be adapted and extract any feature, the DWT coefficients were also obtaine to be tested. In this case a Daubechies 2 wavelet was used, and later the relative energy was obtained to use those coefficients.

In Fig. 4.4 windows are extracted from a sinusoidal signal. As it can be seen, the windows sliding requires more than 3 samples to obtain one period from this signal.

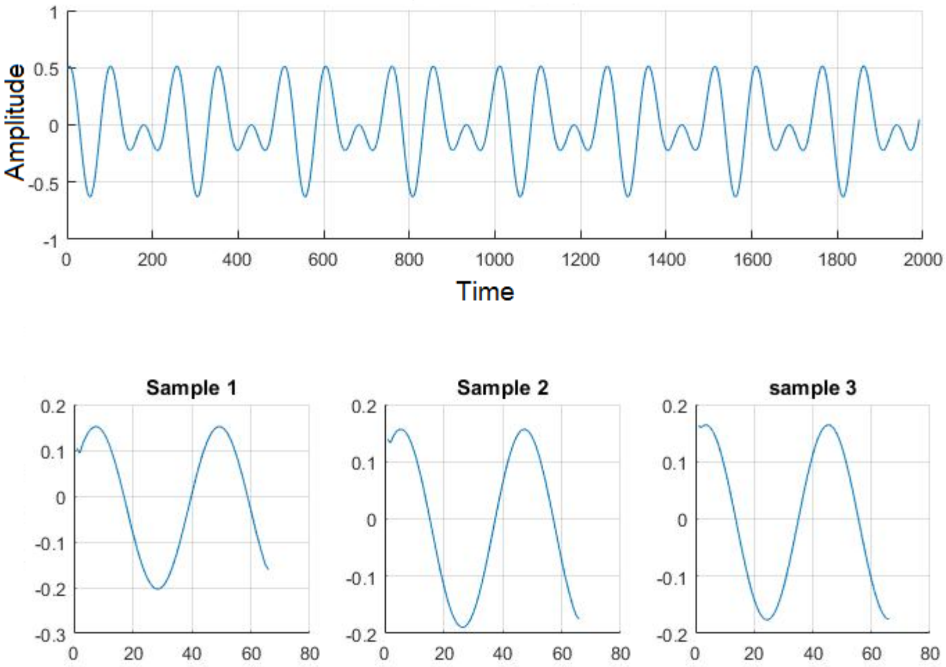


Figure 4.4: Window extraction example

4.4 Clustering

The extracted features are used later to generate a codebook through a clustering method, the objective is to obtain the most representative features or generate prototypes from the existent data. As in [Plinge et al., 2014], the clusters were generated by class and later concatenated in one codebook. The objective is to avoid the mitigation of differences among classes without increasing the clusters number.

The k-means algorithm was used to perform this task, an inconvenience with this method is the previous definition of the clusters number. To define an arbitrary number of clusters it is necessary to have information of the problem and the data, but this implementation has not been applied to a similar dataset, to overcome this problem a genetic algorithm was applied to define a fixed clusters number.

Using a sinusoidal signal, an example codebook can bee seen in Fig. 4.5. In this case, ten codewords were generated to represent the signal showed in Fig. 4.4.

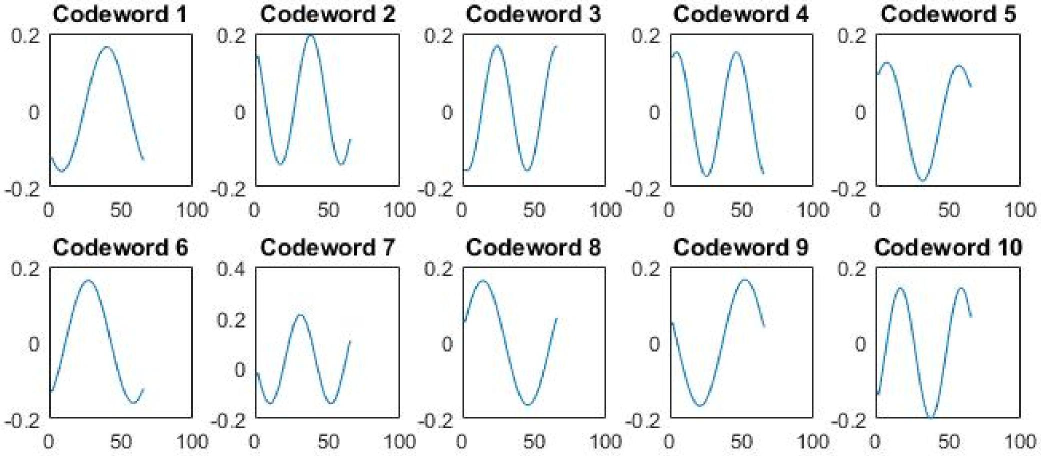


Figure 4.5: Codebook generation example

Once the codebook is generated, the next step is to replace every instance which generate the codebook with a codeword, the result will be a sequence of the codewords over the original data. To choose the codeword which will replace an instance a similarity measure is applied, one of the most simple method is the Nearest Neighbor search and the used measure is the Euclidean distance defined in Eq. 4.2.

$$d_E(P,Q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \tag{4.2}$$

where $P = \{p_1, p_2, ..., p_n\}$ and $Q = \{q_1, q_2, ..., q_n\}$, represent vectors in Euclidean space.

In summary, a codebook is generated by a clustering method, later the original instance values will be replaced by the nearest codeword using the Euclidean distance.

4.5 Histogram generation

At this step, the original signals become sequences of the codebook's codewords, later the occurrence of the codewords is counted in each instance, for this task a convenient representation is a histogram.

Due to each repetition has different length, histograms will have different number of elements. To address this issue the histograms are normalized. Thus, histograms sum of elements are equal to one.

The BoF method itself looses any temporal information of the data, in attempt to consider it, the histogram can be extended to count sequences of two or more codewords, this allows the BoF to consider temporal information of the signal in addition to the main feature extraction step. Counting the occurrence of sequences add to the histogram extended features and make it longer than an ordinary histogram, the sequences are counted if they appear strictly in the same order, this method is called n-grams and was originated in text analysis, in Fig. 4.6 an example of one to three n-grams extraction from a sentence is shown, this idea is applied to codewords sequences.

4.6 Classification

Once the data is converted in a set of histograms, a class (i.e. imagined word) is added to each one of them and become instances for a classifier.

The instances were classified using Multinomial Naive Bayes which was created originally to classify histograms in text analysis. The results were obtained per

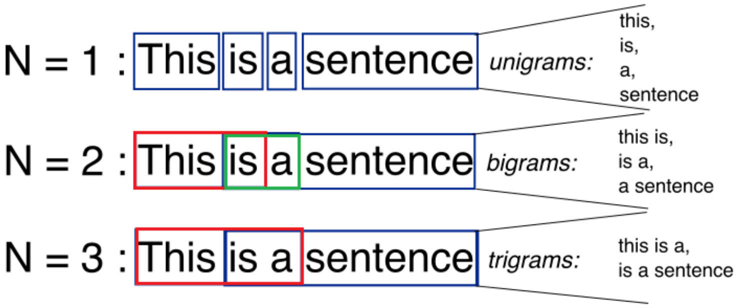


Figure 4.6: n-grams in text analysis [Graber, 2011].

subject and averaged to obtain a general result of the method.

4.7 Summary

Following the general proposed method, chapter 5 will explain in detail the modifications applied to this base model. The first obstacle was to define the parameters that method requires (i.e. extracted features, extraction window size, extraction window sliding and codebook size), to solve this a genetic algorithm was applied.

Also, different signal representations were explored in the feature extraction step, this was a key step because the data is conformed by many channels which represent the activity in different areas of the brain, the idea was to search for patterns in these channels together and individually.

The clustering step must create the most representative prototypes of the imagined words, and the results obtained by means of the genetic algorithm may represent a local maximum, thus, expectation maximization algorithm was applied to find the optimal clustering representation of the data.

Chapter 5

Experiments and results

5.1 Parameters definition

Due to the huge amount of combinations among the BoF parameters, the first approach was to apply a genetic algorithm to find an initial configuration. The parameters to define were: features to extract (C = FFT, DWT), window size to features extraction ($8 \leq W \leq 128$), window sliding ($8 \leq M \leq 128$), and the clusters number ($K \leq 1000$); being a total of four parameters.

Considering this parameters, the worst case will have $28,800,000$ combinations, if intervals of 10 are taken from variables, there will be $28,800$ available combinations. In a computer with an Intel Core i7-3770k processor at 3.5 GHz, 24 GB RAM and 1 Tb HDD; to execute the method ten times for 27 subjects, using Matlab 2017b, takes 833.58 seconds, multiplying it by the combinations number, the result is $28,800 \times 833.58 = 24,007,104$ seconds $= 400,118.4$ minutes $= 6,668.64$ hours $= 277.86$ days. Thus, a genetic algorithm was considered as a first approximation. This parameters can be seen as a vector of features in Fig 5.1.

The objective function of the algorithm was set to minimize the imagined

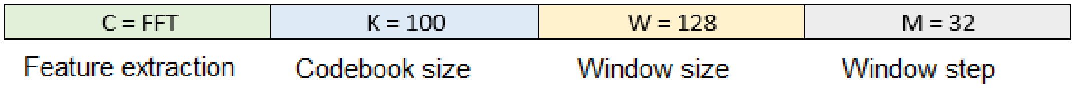


Figure 5.1: Parameters example

speech classification error from all subjects, this error was measured averaging the error from each subject, the five imagined words averaged classification error was used per subject. Eq. 5.1 represent the objective function, where acc is the classification accuracy of the i_{th} subject.

$$\alpha = \min(100 - \frac{\sum_{i=1}^{subjects} acc_i}{subjects}) \tag{5.1}$$

The population size and the generations were fixed to one hundred, the individuals crossover fraction was set to 80% and is given by a weighted averages of the parents. The selection process for reproduction is performed by an uniform stochastic function that moves among the individuals in a fixed step size. Also elitism of 2% WAS considered, this is two individuals per generation that will be conserved to the next generation. The mutation probability for subjects was 1%, the mutated subjects will be modified randomly to diversify the results obtained.

No time or aptitude limits to stop the algorithm were defined, it could continue until the generations were finished ignoring time and the accuracy obtained in each generation. The only stopping criteria was the lack of change in accuracy after twenty generations. In this case the algorithm has reached a stall at generation number seventy one and stopped its execution, taking the best subject at that generation.

The obtained parameters were: a characterization C with FFT analysis, a codebook size K of 75 clusters, a window W of size 40 with a sliding M of 8. This parameters, obtained the results shown in Fig. 5.2.

As expected, the genetic algorithm minimizes the aptitude on each generation,

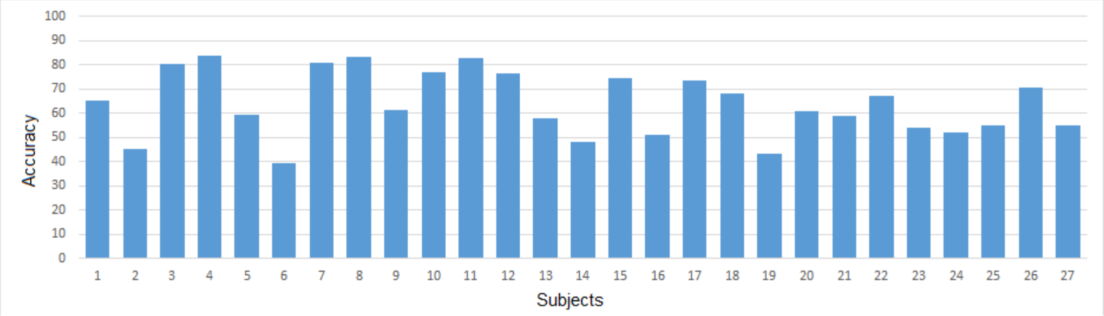


Figure 5.2: Genetic algorithm classification results

in this case, the classification result. The Fig. 5.3 shows the aptitude obtained, and that a stall was reached in generation seventy one where the variation of aptitude was lower than 1×10^{-6}.

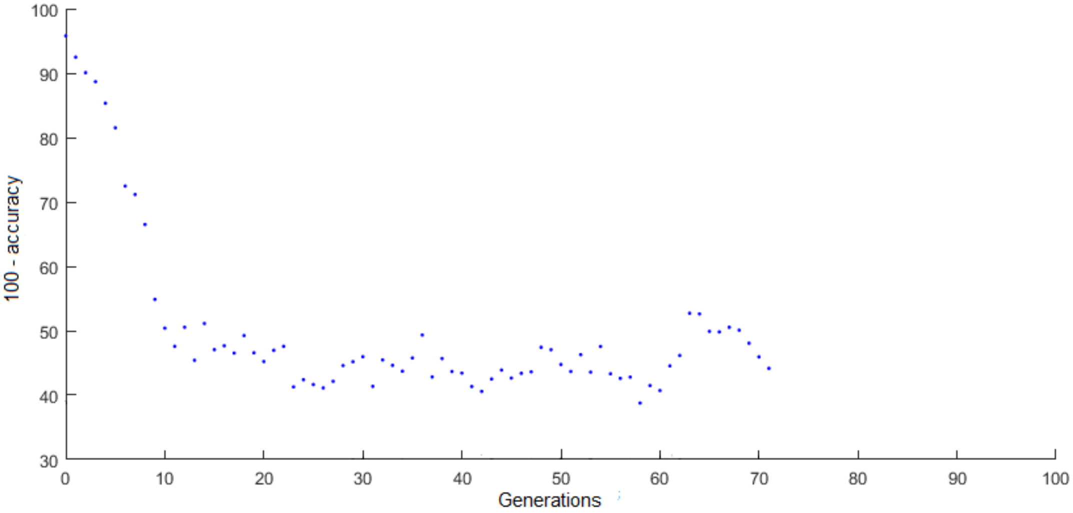


Figure 5.3: Genetic algorithm behavior

Using the parameters resulted from the genetic algorithm, experiments in further sections were realized. This experiments aimed to explore different ways of signal representation. The clustering step was also explored in section 5.3.2 and the first obtained value of 75 clusters were replaced with 200 clusters in the following experiments.

5.2 BoF representations

In this section the feature extraction step will be treated. The feature extraction is of interest due to the fact that it is not clear yet which features are necessary to identify the imagined speech. Different representations were explored based on related works on imagined speech and BoF for time series analysis.

The analyzed representations explore different features of the signal. A first set of representations are based on the frequency analysis of the signal (section 5.2.1, to later include spatial (section 5.2.2) and spatial-temporal (section 5.2.3) information. A second set of experiments combine the electrodes information directly, without a frequency analysis, this is another representation which considers spatial information (section 5.2.4) and later adds temporal information (section 5.2.5).

Each imagined word recording W_i for the S_j subject can be seen as a 14 by n matrix, where 14 is the channels number and n is the samples number (see Fig. 5.4).

$$X_{W_i S_j} = \begin{bmatrix} x_{1,1}, x_{2,1}, ..., x_{14,1} \\ x_{1,2}, x_{2,2}, ..., x_{14,2} \\ \vdots \\ x_{1,n}, x_{2,n}, ..., x_{14,n} \end{bmatrix}$$

Figure 5.4: Signals representation.

The matrix $X_{W_i S_j}$, will be transformed to obtain different representations. These representations use segments $\vec{x}$ from the matrix X to extract features, the FFT function Φ is represented as $\vec{y} = \Phi_c(\vec{x})$, where $c = 1, 2, ..., 14$, and is composed by the following vectors.

$\vec{x} = [x_1, ..., x_w]$, where w is the window size.

$\vec{y} = [y_1, ..., y_n]$, where $\vec{y}$ is the set of FFT coefficients resulting from apply function Φ to a vector $\vec{x}$.

5.2.1 Standard representation

The first approach is based in the standard BoF representation, where each channel is seen as an independent signal, and the extracted windows represent independent instances for the codebook generation. In Fig. 5.5, a description of the representation is shown, where the function $\Phi(\vec{x})$ represent the FFT function for a w size window of the signal, $\vec{x} = [x_1, x_2, ..., x_w]$. This returns a vector which contains the FFT coefficients for a $w = 64$ window size, this is $\vec{y} = \frac{64}{2} + 1 = 33$ coefficients, only half of the coefficients are considered due to the redundancy in the resulting vector.

$$\cup_{subject=1}^{27} \cup_{word=1}^{5}$$
$$\cup_{repetition=1}^{33} \{[\Phi_1(\vec{x_1}), ..., \Phi_1(\vec{x_n})], ..., [\Phi_{14}(\vec{x_1}), ..., \Phi_{14}(\vec{x_n})]\}_{subject,word,repetition}$$

Figure 5.5: Standard representation.

The results per subject of this representation are shown in Fig. 5.6, the experiment was repeated ten times for each subject.

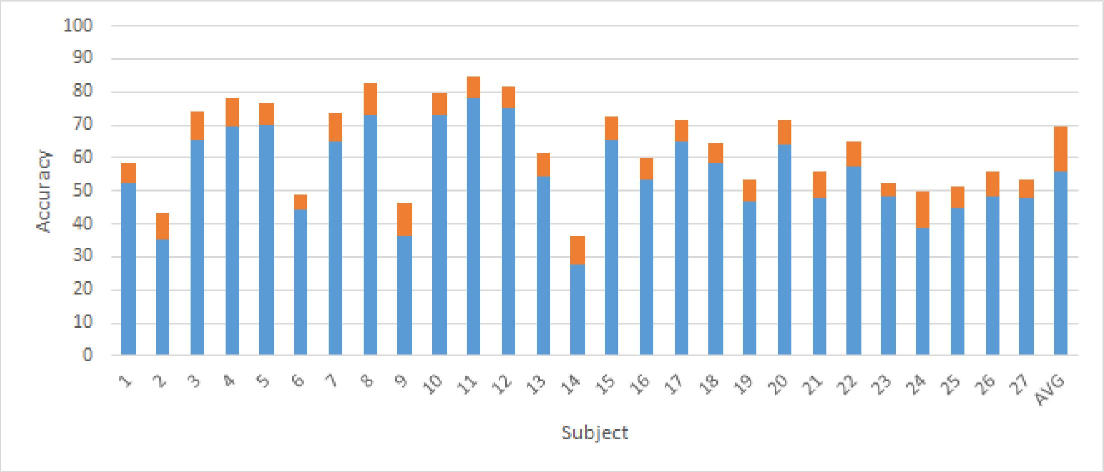


Figure 5.6: Standard representation accuracies in blue, standard deviations in orange.

5.2.2 Windowed spatial representation

Adding spatial information to the representation could allow the detection of different patterns and increase the classification rates. In a different scheme, shown in Fig. 5.5, the extracted windows $\Phi(\vec{x})$ are concatenated by channels adding spatial information of the signals to the representation. The function $\Phi(\vec{x})$ represent the FFT function for a w size window of the signal, and the resulting vectors, formed by FFT coefficients can be calculated as $\vec{y} = (\frac{64}{2} + 1) \times 14 = 462$ coefficients, due to the window size is 64 and the 14 channels are concatenated. Classification results per subject are shown in Fig. 5.8.

$$\cup_{subject=1}^{27} \cup_{word=1}^{5} \cup_{repetition=1}^{33} \{[\Phi_1(\vec{x_1}), ..., \Phi_{14}(\vec{x_1})], ..., [\Phi_1(\vec{x_n}), ..., \Phi_{14}(\vec{x_n})]\}_{subject,word,repetition}$$

Figure 5.7: Windowed spatial representation.

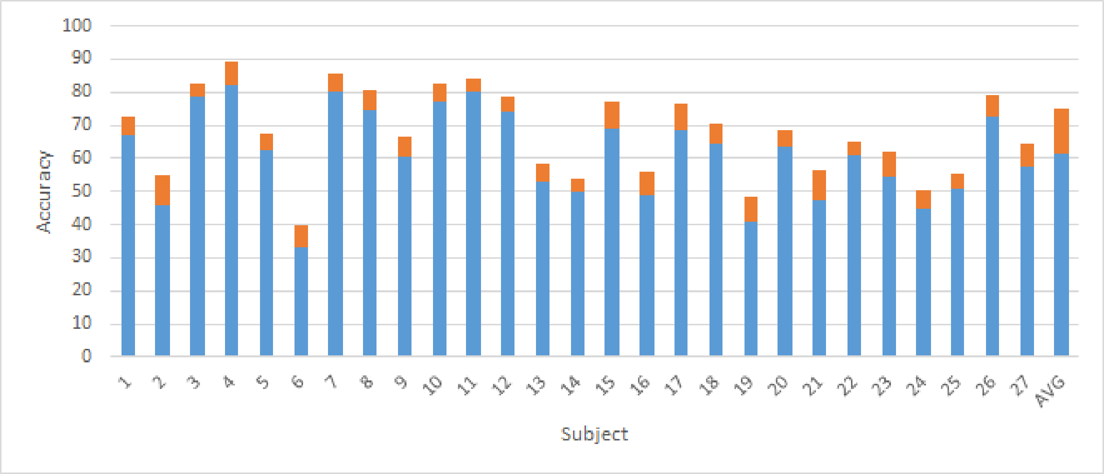


Figure 5.8: Windowed spatial representation accuracies in blue, standard deviations in orange.

5.2.3 Windowed spatial-temporal representation

In an attempt to improve the previous representation, temporal information of the signal is added in histogram generation step, maintaining the other steps as the previous representation. The method implies the use of n-grams in the histogram generation. Based in the previous windowed spatial representation, histograms were generated taking the information of 1-grams to 6-grams, considering the n-grams between them to find temporal information. Classification results per subject are shown in Fig. 5.9.

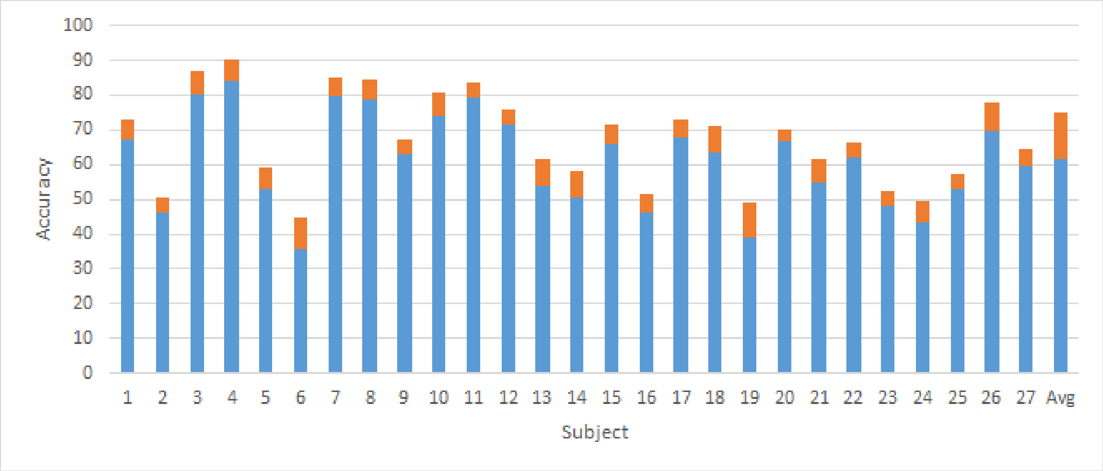


Figure 5.9: Windowed spatial-temporal representation accuracies in blue, standard deviations in orange.

5.2.4 Raw signal spatial representation

A completely different scheme, based on Multivariate Bag of Patterns from [Ordonez et al., 2011], does not apply any feature extraction method and the instances are created using the microvolt values of the channels in every time instant as features, resulting in many instances as samples recorded. The objective of using spatial information of the signal, is to detect patterns of the different cognitive areas of brain and its activation during imagined speech.

In this representation, shown in Fig. 5.10, there is no window feature extraction, the signal microvolt values are used. The instances are made by taking the samples from all channels at the same instant. This representation only requires the definition of the clusters number.

$$\vec{y} = \{[x_{1,1}, ..., x_{14,1}], ..., [x_{1,n}, ..., x_{14,n}]\}$$

Figure 5.10: Raw signal spatial representation without frequency analysis.

Results per subject are shown in Fig. 5.11.

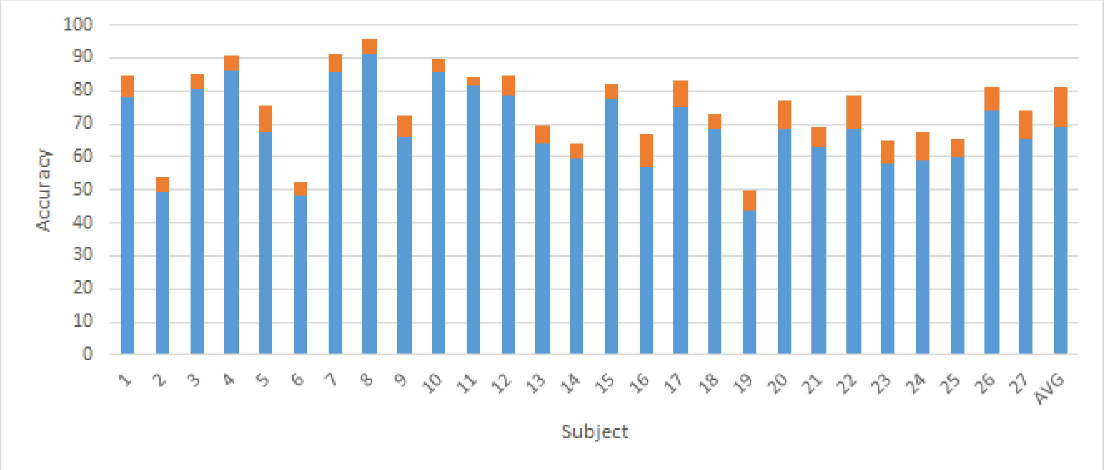


Figure 5.11: Raw signal spatial representation accuracies in blue, standard deviations in orange.

5.2.5 Raw signal spatial-temporal representation

Due to the improvement in raw signal spatial representation, the incorporation of temporal information was tested, this information of the signal is added in histogram generation step, following the previous representation steps. Histograms were generated taking the information of 1-grams to 6-grams, considering the n-grams between them to find temporal information, results are shown in 5.12.

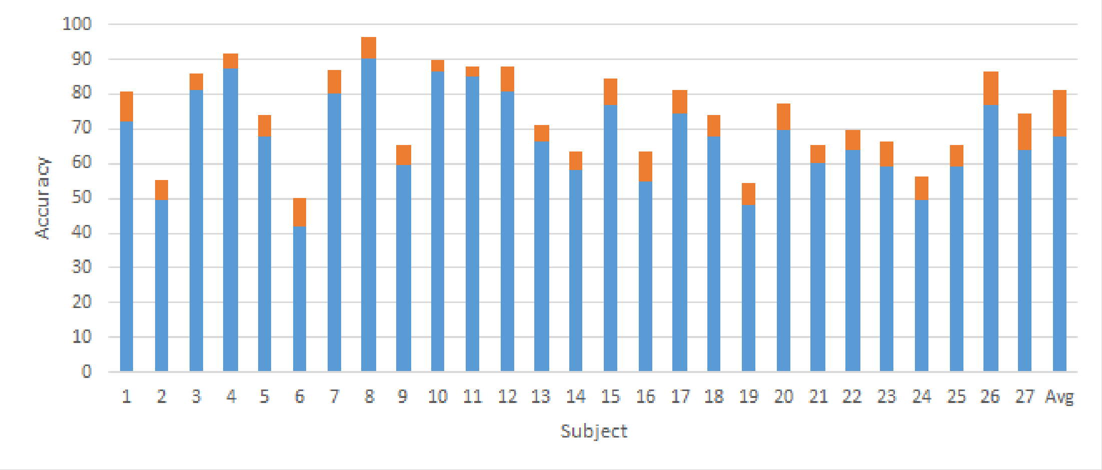


Figure 5.12: Raw signal spatial-temporal representation accuracies in blue, standard deviations in orange.

5.3 Results discussion

5.3.1 Representations

The explored representations have shown similar results (see Table 5.1). Nevertheless, for further tests the raw signal spatial representation was used due to that it shown the best accuracy performance. Raw signal spatial representation detailed results are compared with baseline in Fig. 5.13. In this comparison, can be seen that the proposed method have reached similar results as related work.

The representations can be seen as two sets. The first set includes the Standard, Windowed spatial, and Windowed spatial-temporal representations. And the second set includes Raw signal spatial and Raw signal spatial temporal representations. The difference on these representations lies on the feature extraction and histogram generation steps, the first set includes frequency information and is based on windows. In Table 5.1 the representations results are compared with [Torres-García et al., 2016], which are taken as the baseline results.

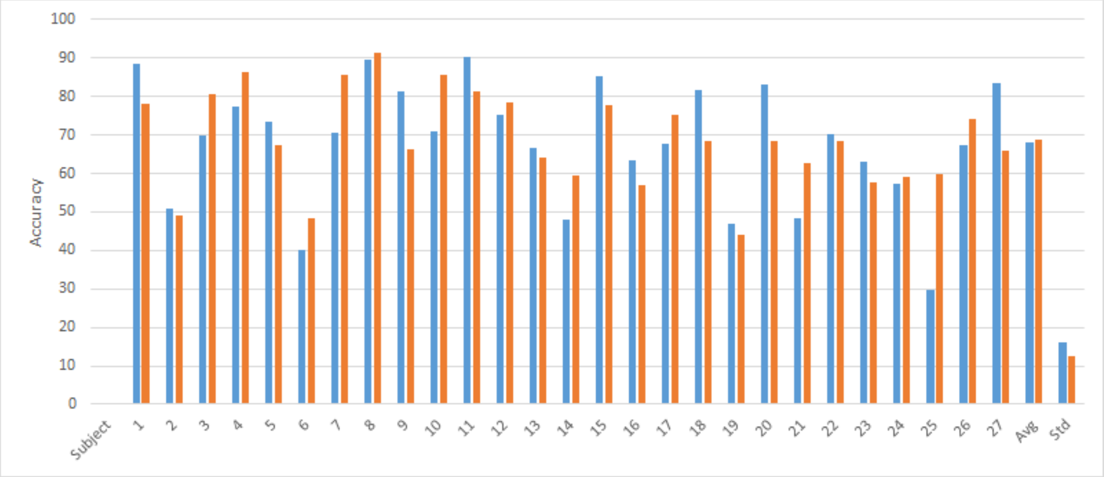


Figure 5.13: Accuracies comparison, [Torres-García et al., 2016] in blue, present work in orange.

The first set of representations was overcame by the second set accuracies, this means that, consider the relation among channels is better to classify imagined speech in this dataset than use each channel as an independent signal. In both representation sets were seen that temporal information on the histogram generation decreases the recognition rate. Nevertheless, it is not rejected that another temporal information extraction methods may achieve better results. It may be due to that obtained temporal information was not able to capture imagined speech intervals.

An interesting result was the increment of recognition rate using raw signal rather than frequency features. Frequency transforms are usually applied to analyze the signal in a lower complex domain. Nevertheless, as was seen in windowed spatial representation, using this information together do not improve the recognition rate, this may be due to this representation is too strict to find patterns in imagined speech for this dataset.

To determine the statistical significance of the representations, a Kruskal-Wallis test was applied. The obtained p-value for the null hypothesis that the data in each method comes from the same distribution was 0.0257, this is a significance level of

Table 5.1: Representations comparison

Number	Representation	Accuracy
1	Standard	$55.77 \pm 13.58\%$
2	Windowed spatial	$61.53 \pm 13.54\%$
3	Windowed spatial-temporal	$61.34 \pm 13.55\%$
4	Raw signal spatial	$68.93 \pm 12.43\%$
5	Raw signal spatial-temporal	$67.78 \pm 13.23\%$
6	[Torres-García et al., 2016]	$68.18 \pm 16\%$

97%. Nevertheless, in Table 5.2, a Dunn-Sidak *pos hoc* test was applied to determine which methods pairs are significantly different resentation numbers are taken from Table 5.1 .

Table 5.2: Dunn-Sidak test p-values

Representations	1	2	3	4	5	6
1		0.59	0.885	0.063	0.11	0.04
2	0.59		1	0.99	0.99	0.99
3	0.88	1		0.94	0.98	0.89
4	0.06	0.99	0.94		1	1
5	0.11	0.99	0.98	1		1
6	0.04	0.99	0.89	1	1	

The obtained p-values show that representation number one is the most different representation. And the remaining methods are statistically similar to each other.

5.3.2 Clustering

A performance of the k-means algorithm with different clusters number is shown in Fig. 5.14, this test was made in the raw signal spatial representation, which does not require more parameters than clusters number. Due to that there is only one parameter to define, a genetic algorithm is not required.

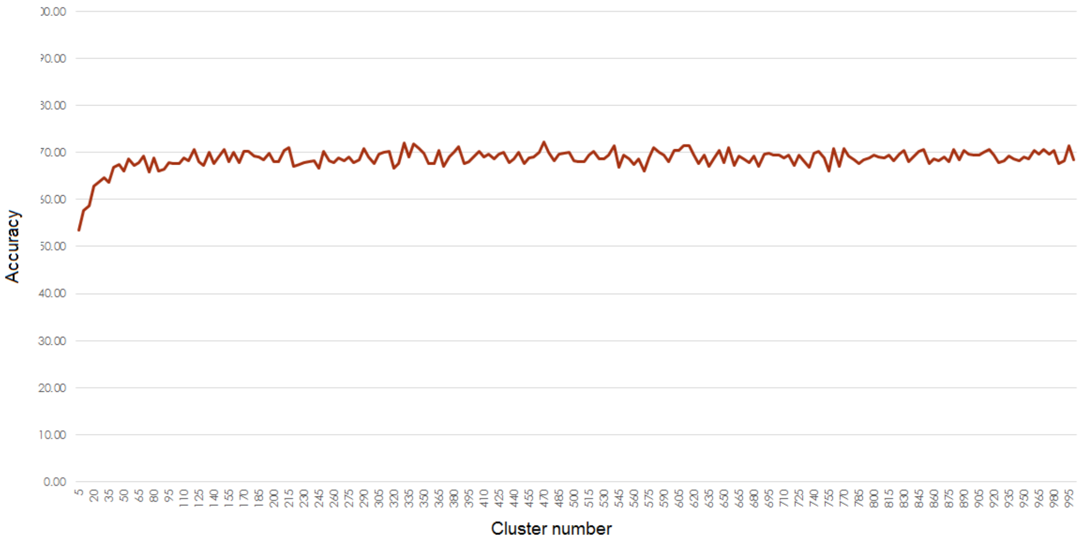


Figure 5.14: Clusters variation

These results were compared with another method, the Expectation Maximization (EM) clustering, which searches for the optimal clusters number, Table 5.3 shows the obtained clusters numbers for ten iterations.

The first cluster definition were given by a genetic algorithm, which is susceptible to stall in a local maximum, also the test results in Fig. 5.14 shows variations in the results, and to choose a cluster number may not be easy. The EM clustering tends to adapt better to the data than k-means does, also, due to it is a probability search based method, it stops when an optimal clustering is done. Thus, the results from Table 5.3 were considered rather than genetic algorithm results for cluster size.

Table 5.3: EM clusters number

Classes	Clusters AVG
Class 1	40.56 ± 17.29
Class 2	41.37 ± 14.40
Class 3	40.15 ± 16.36
Class 4	41.67 ± 17.89
Class 5	36.74 ± 13.22
Sum	200.48
Classes AVG	40.09 ± 1.97

5.3.3 Database testing

The proposed method was tested in different databases to analyze its behavior. First, the database proposed in [Pressel Coretto et al., 2017], using the windowed spatial representation was tested. This database includes 6 imagined Spanish words (i.e. "Arriba", "Abajo", "Izquierda", "Derecha", "Adelante", "Atras"), and was taken fifteen subjects. This database is different in several aspects to the first database used in our work. First, it was taken at a sample rate of 1024 Hz with a six channel EEG system (i.e. F3, F4, C3, C4, P3, P4), this reduce the information that can be extracted from the signal. Second, the acquisition protocol required the subjects to imagine the words three times in a two seconds time lapse. For this experiment the signals were down sampled at $\frac{1}{7}$ of the original.

Table 5.4 shows the results with the proposed representations and the results in [Pressel Coretto et al., 2017] for three subjects. The results improve slightly when the windowed spatial representation is used, this may be due to there are a higher sample rate in the EEG signal. The spatial representation shown lower results than related work, it is assumed that it was caused by the low amount of channels, which also are unrelated to speech area. It is important to recall that the acquisition protocol record the repetition of words several times and this may have an impact in codewords generation.

Table 5.4: Accuracy comparison with [Pressel Coretto et al., 2017]

Subject	Windowed Spatial Representation	Spatial Representation	Pressel
1	20.93 ± 1.4	14.16 ± 3.26	19.31
2	21.79 ± 2.39	14.09 ± 2.47	19.58
3	21.31 ± 1.59	16.51 ± 3.15	19.92
AVG	21.35 ± 0.43	14.92 ± 1.37	19.6 ± 0.3

Moreover, a Kruskal-Wallis test among Windowed-Spatial Representation and [Pressel Coretto et al., 2017] were applied. The obtained p-value was 0.0495, thus, a significance level of 95% was achieved among these methods.

In [Zhao and Rudzicz, 2015], another database of imagined speech is presented, this database has four imagined words (i.e. ”pat”, ”pot”, ”knew”, ”gnaw”), with a 64 channel EEG system at 1000 Hz. Nevertheless, the classification in this work was realized among phonological categories and mental states, thus, there is no comparison with present work, nevertheless the classification results of imagined words using the raw signal spatial representation are shown in Table 5.5.

The classification results were very variable among subjects, some results (i.e. subject 9) do not overcame results by chance, but some others (i.e. subject 12) shown higher results. Nevertheless, considering the average among subjects, the method needs a deeper analysis to be adapted to this database.

Table 5.5: Raw signal spatial representation results in Zhao database

Subject	Accuracy
1	35 ± 13.56
2	31.66 ± 6.57
3	34.16 ± 9.17
4	43.33 ± 14.5
5	34.16 ± 12.6
6	36.66 ± 8.95
7	27.5 ± 10.43
8	30 ± 10.54
9	15 ± 9.46
10	19.16 ± 6.86
11	29.16 ± 13.17
12	55.83 ± 8.82
13	23.33 ± 8.60
14	26.25 ± 7.68
AVG	31.51 ± 10.11

5.4 Transfer learning

5.4.1 Imagined words

Using the raw signal spatial representation, a different experiment was proposed. The objective is to test the method in a vocabulary expansion scenario, without training the model for a new class (i. e. a new word). In this experiment one class was excluded from the codebook generation, and adding it to the classification step. The experiment was repeated for each class. For the results comparison in Table 5.6

a confusion matrix of the results is shown, this will help as a comparison with the following matrices where one word will be tested at once.

Table 5.6: Base line confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
74.95	9.76	4.72	6.06	4.49	Arriba
8.33	67.26	6.8	13.37	4.21	Abajo
3	7.36	66.34	10.74	12.254	Izquierda
4.62	13.19	12.5	62.17	7.5	Derecha
3.19	6.57	12.54	4.76	72.91	Seleccionar

Class ”Arriba” shown an accuracy decrease of 14.35% when is excluded from the codebook generation, nevertheless , class ”Abajo” increase slightly its recognition rate. Table 5.7 shows the confusion matrix for the class ”Arriba” exclusion. Other classes also shown a decrease on the recognition rate with a mean of 63.98% for all classes, this is a decrease of 4.95% when no transfer learning is applied, less than class ”Arriba” recognition decrement.

Table 5.7: Word ”Arriba” transferring confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
60.6	18	11.15	6.15	4.07	Arriba
14.21	69.67	6.66	5.18	4.25	Abajo
11.66	12.03	59.58	10.5	6.2	Izquierda
5.69	10.78	9.62	60.8	13.05	Derecha
6.38	9.07	4.12	11.15	69.25	Seleccionar

Class ”Abajo” shown an accuracy increase of 3.2% when is excluded from the codebook generation, also class ”Izquierda” and ”Derecha” increase slightly them recognition rate. The mean recognition rate was 65.11% for all classes, this is a decrease of 3.81% when no transfer learning is applied. Table 5.8 shows the confusion matrix for the class ”Abajo”.

Table 5.8: Word ”Abajo” transferring confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
68.05	7.5	12.36	8.05	4.02	Arriba
17.22	70.46	4.3	4.49	3.51	Abajo
22.36	3.84	56.38	10.41	6.99	Izquierda
14.49	2.03	8.65	61.89	12.91	Derecha
12.77	3.24	3.28	11.8	68.8	Seleccionar

Class ”Izquierda” shown an accuracy decrease of 7.78% when is excluded from the codebook generation, classes ”Arriba” and ”Seleccionar” also shown an slightly decrease on them recognition rate, in other hand, classes ”Abajo” and ”Derecha” shown a small increment. Mean recognition rate was 65.68% for all classes, this is a decrease of 3.25% when no transfer learning is applied. Table 5.9 shows the confusion matrix for the class ”Izquierda”.

Table 5.9: Word ”Izquierda” transferring confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
64.02	6.2	11.75	14.39	3.6	Arriba
10.69	71.29	5.97	8.65	3.37	Abajo
10.55	3.88	58.56	21.11	5.87	Izquierda
7.03	2.96	10.55	67.91	11.52	Derecha
5.09	3.47	3.47	21.29	66.66	Seleccionar

Class ”Derecha” shown an accuracy decrease of 3.01% when is excluded from the codebook generation, classes ”Arriba”, ”Izquierda” and ”Seleccionar” also shown an slightly decrease on them recognition rate. Mean recognition rate was 65.06% for all classes, this is a decrease of 3.86% when no transfer learning is applied. Table 5.10 shows the confusion matrix for the class ”Derecha”.

Table 5.10: Word ”Derecha” transferring confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
63.24	6.01	21.06	5.09	4.58	Arriba
10.69	70.87	10.97	4.12	3.33	Abajo
13.61	4.9	63.37	10.32	7.77	Izquierda
5.6	2.26	20.69	59.16	12.26	Derecha
5.32	3.84	11.85	10.27	68.7	Seleccionar

Class ”Seleccionar” shown an accuracy decrease of 1.39% when is excluded from the codebook generation, nevertheless, classes ”Arriba”, ”Abajo” and ”Seleccionar” also shown an slightly decrease on them recognition rate. Mean recognition rate was 63.97% for all classes, this is a decrease of 4.95% when no transfer learning is applied. Table 5.11 shows the confusion matrix for the class ”Seleccionar”.

Table 5.11: Word ”Seleccionar” transferring confusion matrix percentages

Arriba	Abajo	Izquierda	Derecha	Seleccionar	← Classified as
64.39	5.83	11.62	5.83	2.31	Arriba
11.99	70.5	6.01	3.51	7.96	Abajo
11.52	3.47	57.5	9.35	18.14	Izquierda
6.94	1.94	10.04	55.97	25.09	Derecha
5.87	3.6	6.06	12.918	71.52	Seleccionar

As can be seen, each word which is excluded, affects the other words in different ways, this may be related to the words semantic similarities. Nevertheless, in general, the classification accuracy has decreased 4.16% when transfer learning is applied, it is important to recall that the method did not have any information of the excluded word for the codebook generation.

5.4.2 Subjects Leave-One Out

Following the previous idea, this time the codebook was generated for a subset of subjects, and tested for the rest of them. This can be seen as a Leave-One Validation scheme, but it allows to test the trained model in transfer learning among different subjects.

The first seventeen subjects were selected to generate the codebook using the spatial representation. The representation was tested ten times for each one of the remaining subjects. Table 5.12 shows the average accuracy per subjects.

Table 5.12: Subjects exclusion accuracies

Subject	Accuracy
18	21.63 ± 2.32
19	19.81 ± 2.32
20	14.66 ± 2.84
21	22.48 ± 1.75
22	27.45 ± 2.86
23	25.69 ± 2.36
24	16.18 ± 2.46
25	15.45 ± 1.32
26	22.66 ± 1.92
27	25.21 ± 4.04
Avg	21.13 ± 4.5

Obtained results have shown a low classification accuracy using the proposed method, in fact, the results are close to the chance probability of 20%. A proposed solution is to redefine the parameters (i.e. cluster number, proposed representation).

To solve this issue a calibration step could be added, in which few samples of the new subjects are included in the codebook generation. This calibration is commonly used in BCIs, but the experiment objective was to make a general codebook representation which be able to adapt to any subject.

Moreover, the representation depends on the acquisition protocol, as well on the acquisition device. Thus, this experiment must be realized over different databases.

Chapter 6

Conclusions and future work

6.1 Conclusions

In this work, different novel representations for imagined speech classification were explored, applied in a bag of features method. The method was improved following schemes used in diverse areas, spatial, temporal and frequency features were considered to find the best BoF model for the problem.

The obtained results have shown that it is possible to detect patterns of imagined speech in EEG signals using the BoF method, the best model was a raw signal spatial representation (section 5.2.4) which obtained an average accuracy of 68.93 ± 12.43 for all subjects.

It was seen that the use of temporal information on histograms generation does not improve the performance of the method, even, in some cases it reduces the classification rates. First assumption is that the detection of changes at the actual sample rate is not adequate to imagined speech. Another assumption is that the requirement of n-gram sequences added to the raw signal spatial representation is very strict to pattern detection.

The use of the microvolt signal values in the raw signal spatial representation gives information of the activation areas over the scalp, in contrast the use of FFT and DWT coefficients reduced the obtained results. Moreover, the windowed spatial representation, which included parts of both representations does not improved the results.

The attempt of extend the method to different databases shown encouraging results, a detailed adjustment to each database may improve the obtained results, it is important to remark that a defined standard for imagined speech database creation does not exist and every work generates its acquisition protocol according to its specific problem. This is an important issue to do a method comparison. Nevertheless this is a challenge that must be overcame, due to that in the real applications users must be able to use their natural imagined speech with out protocols.

Transfer learning experiments obtained interesting results when new words are added to the vocabulary with a mean decreasing of 4.16 in the recognition rates accuracies. The confusion matrices, as expected, showed errors in attempt to classify the new word, this could be solved with a calibration step, but in this case the objective was to analyze the behavior of the method without any information of the new word.

6.2 Future work

This work can be improved in different steps following the next proposals. In the feature extraction step, different methods could be applied, as the Matching Pursuit, which is called as a generalization of the DWT [Mallat and Zhang, 1993]. The genetic algorithm used for a first approximation could be extended to search for more parameters (e.g. wavelet families or decomposition levels).

The features extraction step could consider the EEG bands related to cognitive activities known in literature [Zarate, 2005], this could lead to a different representation of the extracted features. Nevertheless, to remove EEG bands is not considered yet, because an analysis of which bands contributes to imagined speech needs to be realized first, a first approach could consider all bands.

In clustering step a cluster analysis can be realized, where the cluster cohesion is used as a measure to select the most joint clusters. For this task a threshold needs to be defined, it was fixed in the clusters cohesion mean minus one standard deviation. Another measure that complements the cohesion, is the cluster separation, a set of rules could be defined using both measures to select the most representative clusters. Nevertheless, this could lead to suppress clusters which can differentiate among classes.

The histogram generation step can be extended to uses different text analysis techniques, one of them can be the skip grams method, this is a generalization of the n-grams which can create sequences that are not consecutive in the signal. This representation may be able to found new patterns which were not considered in present representations.

The method could be extended to different brain activities (e.g. imagined movement), pathologies or biometric databases. Nevertheless, the method requires to be adapted to the databases, because each one is taken with different environments variables, protocols and devices.

Also different processing methods could be applied to extract signal features. These features could be related with brain internal processes. In appendix C a dipole analysis is explored using the proposed method, this analysis tries to identify the activation areas of the brain for a specific activity. This approach attempts to explore in the inner brain function of imagined speech, which is a neuro-science related problem, but may help in the imagined speech classification.

In transfer learning experiments, a calibration step may help to increase the recognition rates, this consists in add a small amount of the new word instances in the codebook generation step. Including this data, the model may adapt better to the new world, the advantage is that this step requires few instance of the new word, in practice this is few time of the user training in a previous trained model.

Appendix A

Representations detailed results

In this appendix, the results per subject are shown. This experiments were repeated 10 times.

Table A.1 shows the results obtained from standard representation, see Section 5.2.1.

Table A.2 shows the results obtained from windowed spatial representation, see Section 5.2.2.

Table A.3 shows the results obtained from windowed spatial-temporal representation, see Section 5.2.3.

Table A.4 shows the results obtained from raw signal representation, see Section 5.2.4.

Table A.5 shows the results obtained from raw signal representation, see Section 5.2.5.

Table A.1: Standard representation accuracies

Subject	Accuracy	Subject	Accuracy
1	52.25 ± 6.06	15	65.75 ± 6.67
2	35.00 ± 8.25	16	53.25 ± 6.67
3	65.75 ± 8.25	17	65.00 ± 6.77
4	69.50 ± 8.48	18	58.50 ± 6.15
5	70.25 ± 6.40	19	46.75 ± 6.57
6	44.25 ± 4.72	20	63.75 ± 7.75
7	64.75 ± 9.01	21	47.75 ± 8.03
8	73.25 ± 9.21	22	57.25 ± 7.77
9	36.25 ± 10.09	23	48.25 ± 4.09
10	73.00 ± 6.65	24	38.75 ± 11.32
11	78.25 ± 6.57	25	44.75 ± 6.40
12	75.00 ± 6.77	26	48.50 ± 7.28
13	54.50 ± 6.95	27	47.75 ± 5.58
14	27.75 ± 8.70	AVG	55.77 ± 13.58

Table A.2: Windowed spatial representation accuracies

Subject	Accuracy
1	67.25 ± 5.46
2	45.75 ± 9.13
3	78.75 ± 3.95
4	82.25 ± 6.92
5	62.25 ± 5.20
6	33.25 ± 6.57
7	80 ± 5.65
8	74.5 ± 5.99
9	60.5 ± 5.87
10	77.25 ± 5.58
11	80 ± 4.41
12	74 ± 4.74
13	52.75 ± 5.46
14	49.75 ± 4.16
15	69 ± 8.01
16	49 ± 6.79
17	68.75 ± 8.10
18	64.5 ± 6.10
19	40.75 ± 7.46
20	63.25 ± 5.41
21	47.25 ± 9.01
22	60.75 ± 4.42
23	54.25 ± 7.64
24	45 ± 5.27
25	51 ± 4.59
26	72.5 ± 6.87
27	57.25 ± 7.40
AVG	61.54 ± 13.54

Table A.3: Windowed spatial-temporal representation accuracies

Subject	Accuracy	Subject	Accuracy
1	67.25 ± 5.95	15	66 ± 5.55
2	46 ± 4.59	16	46 ± 5.68
3	80 ± 6.77	17	67.75 ± 5.46
4	84 ± 6.37	18	63.25 ± 8
5	52.75 ± 6.4	19	39.25 ± 9.86
6	35.5 ± 9.34	20	67 ± 3.29
7	79.5 ± 5.63	21	55 ± 6.67
8	78.75 ± 5.8	22	62.25 ± 4.16
9	62.75 ± 4.48	23	48 ± 4.53
10	74 ± 6.48	24	43.25 ± 6.57
11	79.25 ± 4.09	25	53 ± 4.38
12	71.75 ± 3.92	26	69.75 ± 8.29
13	54 ± 7.38	27	59.5 ± 5.11
14	50.75 ± 7.36	AVG	61.34 ± 13.55

Table A.4: Raw signal spatial representation accuracies

Subject	Accuracy
1	78 ± 6.54
2	49.25 ± 4.72
3	80.5 ± 4.83
4	86.25 ± 4.45
5	67.5 ± 8.16
6	48.5 ± 4.12
7	85.75 ± 5.41
8	91.25 ± 4.29
9	66.25 ± 6.26
10	85.75 ± 4.09
11	81.5 ± 2.69
12	78.5 ± 6.26
13	64 ± 5.55
14	59.5 ± 4.38
15	77.75 ± 4.48
16	57 ± 9.78
17	75.25 ± 7.95
18	68.5 ± 4.74
19	44 ± 5.92
20	68.5 ± 8.6
21	62.75 ± 6.29
22	68.5 ± 9.94
23	57.75 ± 7.12
24	59 ± 8.35
25	60 ± 5.27
26	74 ± 7.09
27	65.75 ± 8.25
AVG	68.94 ± 12.43

Table A.5: Raw signal spatial-temporal representation accuracies

Subject	Accuracy	Subject	Accuracy
1	72.25 ± 8.37	15	76.75 ± 7.82
2	49.75 ± 5.71	16	54.75 ± 8.70
3	81.00 ± 4.89	17	74.5 ± 6.85
4	87.25 ± 4.32	18	68 ± 5.75
5	67.75 ± 6.29	19	48 ± 6.54
6	41.75 ± 8.42	20	69.5 ± 7.80
7	80.25 ± 6.50	21	60 ± 5.27
8	90.25 ± 6.29	22	63.75 ± 5.80
9	59.75 ± 5.46	23	59 ± 7.38
10	86.25 ± 3.58	24	49.5 ± 6.95
11	85 ± 2.64	25	59 ± 6.15
12	80.75 ± 6.98	26	76.75 ± 9.72
13	66.25 ± 5.03	27	64 ± 10.55
14	58.25 ± 5.28	AVG	67.78 ± 13.23

Appendix B

Transfer learning detailed results

This appendix extends the results obtained in transfer learning. This experiments were repeated 10 times.

Table B.1 shows the obtained accuracies per subject excluding the word ”Arriba”.

Table B.2 shows the obtained accuracies per subject excluding the word ”Abajo”.

Table B.3 shows the obtained accuracies per subject excluding the word ”Izquierda”.

Table B.4 shows the obtained accuracies per subject excluding the word ”Derecha”.

Table B.5 shows the obtained accuracies per subject excluding the word ”Seleccionar”.

Table B.1: Word ”Arriba” exclusion accuracies

Subject	Accuracy	Subject	Accuracy
1	73 ± 7.62	15	71.25 ± 6.59
2	44.5 ± 7.89	16	52.5 ± 7.55
3	78 ± 7.15	17	69 ± 4.12
4	83.5 ± 5.55	18	61.5 ± 4.74
5	57.5 ± 7.17	19	41 ± 5.30
6	43.75 ± 8.84	20	65 ± 5.14
7	75.5 ± 10.98	21	54.75 ± 6.29
8	86.5 ± 4.12	22	65.75 ± 6.57
9	60.75 ± 8.08	23	57.75 ± 6.71
10	78.75 ± 5.43	24	57 ± 7.80
11	82.75 ± 3.22	25	64 ± 3.94
12	77.75 ± 7.12	26	69 ± 5.92
13	62.75 ± 7.59	27	55.5 ± 8.8
14	48.5 ± 5.03	AVG	64.35 ± 12.64

Table B.2: Word ”Abajo” exclusion accuracies

Subject	Accuracy	Subject	Accuracy
1	67.25 ± 8.7	15	75 ± 6.77
2	48 ± 7.9	16	52.75 ± 9.16
3	79.25 ± 1.69	17	71 ± 7.19
4	86.75 ± 5.41	18	67.5 ± 3.54
5	66.25 ± 7.10	19	42.75 ± 4.48
6	47.5 ± 6.24	20	69.75 ± 8.29
7	79 ± 6.03	21	60.75 ± 6.13
8	87.25 ± 5.46	22	69 ± 6.48
9	60.75 ± 7.73	23	65 ± 6.67
10	77.5 ± 4.56	24	50.75 ± 5.53
11	80.75 ± 6.24	25	58.75 ± 8.10
12	80.5 ± 5.24	26	76.5 ± 11.32
13	64 ± 5.16	27	61.5 ± 9.22
14	51.5 ± 9.94	AVG	66.56 ± 12.37

Table B.3: Word ”Izquierda” exclusion accuracies

Subject	Accuracy	Subject	Accuracy
1	73.25 ± 5.14	15	73 ± 2.84
2	48.75 ± 5.68	16	56.75 ± 6.35
3	76.75 ± 6.13	17	73 ± 7.62
4	84.25 ± 6.35	18	67 ± 4.22
5	68.25 ± 8.66	19	41.75 ± 6.88
6	46.5 ± 5.68	20	70.75 ± 7.46
7	82 ± 7.53	21	55.5 ± 7.53
8	86.75 ± 4.72	22	65.25 ± 7.02
9	57.5 ± 7.36	23	53.25 ± 5.41
10	81.25 ± 5.56	24	44 ± 7.47
11	85 ± 4.41	25	49 ± 8.83
12	74 ± 5.16	26	73.75 ± 9.30
13	64 ± 9.52	27	55.75 ± 9.43
14	51 ± 6.03	AVG	65.11 ± 13.61

Table B.4: Word ”Derecha” exclusion accuracies

Subject	Accuracy
1	70.25 ± 8.12
2	47.5 ± 6.45
3	80.25 ± 4.32
4	84.75 ± 2.75
5	58.25 ± 6.13
6	44 ± 8.35
7	81.25 ± 6.80
8	88 ± 3.87
9	57 ± 4.05
10	81.5 ± 6.37
11	79.75 ± 5.33
12	76 ± 6.03
13	64.25 ± 6.88
14	54.25 ± 6.46
15	75.75 ± 6.02
16	57.75 ± 6.71
17	72 ± 4.38
18	64.75 ± 6.17
19	40.25 ± 7.59
20	68 ± 4.97
21	60.25 ± 7.12
22	66.75 ± 8.58
23	53.75 ± 8.84
24	52.75 ± 8.03
25	47.5 ± 3.33
26	72 ± 9.34
27	59.25 ± 9.28
AVG	65.10 ± 13.24

Table B.5: Word ”Seleccionar” exclusion accuracies

Subject	Accuracy	Subject	Accuracy
1	71.75 ± 5.78	15	75.25 ± 7.59
2	46.25 ± 6.04	16	51 ± 6.79
3	80.25 ± 4.92	17	74.25 ± 6.88
4	82.25 ± 4.78	18	68.75 ± 6.37
5	61.75 ± 6.35	19	42.25 ± 5.33
6	43.75 ± 6.80	20	69.5 ± 6.65
7	81.75 ± 3.55	21	56 ± 5.80
8	80.5 ± 4.68	22	65.5 ± 6.54
9	57.75 ± 6.29	23	58 ± 6.43
10	79.25 ± 5.90	24	57 ± 8.32
11	73 ± 4.83	25	56.5 ± 4.28
12	74.25 ± 5.90	26	72.5 ± 8.08
13	54.5 ± 4.97	27	54.75 ± 9.24
14	55.75 ± 3.13	AVG	64.59 ± 12.21

Appendix C

Dipole analysis

This appendix introduce the first approach to a generators analysis o dipole analysis. The dipoles of the signals were extracted in different databases and the Raw signal spatial method was tested using this features.

A major obstacle to using EEG data to visualize macroscopic brain dynamics is the under-determined nature of the inverse problem: Given an EEG scalp distribution of activity observed at given scalp electrodes, any number of brain source distributions can be found that would produce it. This is because there are any number of possible brain source area pairs or etc. that, jointly, add (or subtract) nothing to the scalp data. Therefore, solving this 'EEG inverse' problem uniquely requires making additional assumptions about the nature of the source distributions. A computationally tractable approach is to find some number of equivalent current dipoles whose summed projections to the scalp most nearly resemble the observed scalp distribution [Delorme and Makeig, 2004].

This dipoles were calculated using EEGLAB [Delorme and Makeig, 2004]. Table C.1 show the accuracies obtained for [Torres-García et al., 2016] database. Table C.2 show the accuracies obtained for [Zhao and Rudzicz, 2015] database. And Table

C.3 show the accuracies obtained for [Pressel Coretto et al., 2017] database.

Table C.1: [Torres-García et al., 2016] dipole analysis

Subject	Accuracy	Subject	Accuracy
1	22 ± 2.84	15	21.75 ± 4.72
2	24.5 ± 4.68	16	17 ± 4.97
3	23.25 ± 4.42	17	24.5 ± 6.65
4	27.5 ± 4.86	18	23.75 ± 4.75
5	28 ± 4.83	19	29.75 ± 6.06
6	19.75 ± 5.46	20	23 ± 6.85
7	27 ± 3.69	21	21.25 ± 5.56
8	26.5 ± 3.57	22	20.5 ± 5.11
9	24.5 ± 5.99	23	25.75 ± 3.92
10	23.75 ± 5.30	24	21.25 ± 4.29
11	36.5 ± 7.28	25	21.5 ± 2.93
12	21.5 ± 3.57	26	34.25 ± 4.72
13	26.25 ± 4.75	27	29.5 ± 7.43
14	22.5 ± 5.27	AVG	24.71 ± 4.31

Table C.2: [Zhao and Rudzicz, 2015] dipole analysis

Subject	Accuracy
1	27.5 ± 8.94
2	20 ± 7.03
3	14.17 ± 11.15
4	23.33 ± 13.49
5	24.17 ± 8.29
6	36.67 ± 9.78
7	12.5 ± 11.95
8	10 ± 10.24
9	21.67 ± 8.96
10	40.83 ± 10.72
11	26.67 ± 10.24
12	26.67 ± 10.97
13	23.33 ± 5.27
14	24.38 ± 12.99
AVG	23.71 ± 8.4

Table C.3: [Pressel Coretto et al., 2017] dipole analysis

Subject	Accuracy
1	21.11 ± 3.14
2	15.00 ± 3.83
3	19.85 ± 4.36
AVG	18.65 ± 3.23

Bibliography

[Alhaddad, 2012] Alhaddad, M. J. (2012). Common average reference (car) improves p300 speller.

[Arthur and Vassilvitskii, 2007] Arthur, D. and Vassilvitskii, S. (2007). K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '07, pages 1027–1035, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

[Baydogan et al., 2013] Baydogan, M., Runger, G., and Tuv, E. (2013). A bag-of-features framework to classify time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2796–2802.

[CDS, 2017] CDS (2017). http://www.ece.uvic.ca/ elec499/2004a/group05/html/background.html

[Cooley and Tukey, 1965] Cooley, J. W. and Tukey, J. W. (1965). An algorithm for the machine calculation of complex fourier series. *Mathematics of Computation*, 19(90):297–301.

[da Silva, 2009] da Silva, F. L. (2009). EEG: Origin and Measurement. In Mulert, C. and Lemieux, L., editors, *EEG - fMRI*, pages 19–38. Springer Berlin Heidelberg, Berlin, Heidelberg. DOI: 10.1007/978-3-540-87919-0_2.

[DaSalla et al., 2009] DaSalla, C. S., Kambara, H., Sato, M., and Koike, Y. (2009). Single-trial classification of vowel speech imagery using common spatial patterns. *Neural Networks*, 22(9):1334 – 1339. Brain-Machine Interface.

[Dellaert, 2002] Dellaert, F. (2002). The expectation maximization algorithm. Technical report, Georgia Institute of Technology.

[Delorme and Makeig, 2004] Delorme, A. and Makeig, S. (2004). Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of Neuroscience Methods*, 134(1):9 – 21.

[Dewan, 1967] Dewan, E. M. (1967). Occipital alpha rhythm eye position and lens accommodation. *Nature*, 214:975–977.

[EMOTIV, 2017] EMOTIV (2017). https://www.emotiv.com/.

[Farwell and Donchin, 1988] Farwell, L. and Donchin, E. (1988). Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and Clinical Neurophysiology*, 70(6):510 – 523.

[Frigo and Johnson, 2005] Frigo, M. and Johnson, S. G. (2005). The design and implementation of fftw3. *Proceedings of the IEEE*, 93(2):216–231.

[Graber, 2011] Graber, J. B. (2011). Language models.

[Gui and Yeh, 2014] Gui, Z.-W. and Yeh, Y.-R. (2014). *Time Series Classification with Temporal Bag-of-Words Model*, pages 145–153. Springer International Publishing, Cham.

[Heideman et al., 1984] Heideman, M., Johnson, D., and Burrus, C. (1984). Gauss and the history of the fast fourier transform. *IEEE ASSP Magazine*, 1(4):14–21.

[Jin and Han, 2010] Jin, X. and Han, J. (2010). *Expectation Maximization Clustering*, pages 382–383. Springer US, Boston, MA.

[Kim et al., 2013] Kim, T., Lee, J., Choi, H., Lee, H., Kim, I. Y., and Jang, D. P. (2013). Meaning based covert speech classification for brain-computer interface based on electroencephalography. In *Neural Engineering (NER), 2013 6th International IEEE/EMBS Conference on*, pages 53–56.

[Klonowski, 2009] Klonowski, W. (2009). Everything you wanted to ask about eeg but were afraid to get the right answer. *Nonlinear Biomedical Physics*, 3(1):1 – 5.

[Lazebnik and Raginsky, 2009] Lazebnik, S. and Raginsky, M. (2009). Supervised learning of quantizer codebooks by information loss minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(7):1294–1309.

[Lin et al., 2007] Lin, J., Keogh, E., Wei, L., and Lonardi, S. (2007). Experiencing sax: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144.

[Lin and Li, 2009] Lin, J. and Li, Y. (2009). *Finding Structural Similarity in Time Series Data Using Bag-of-Patterns Representation*, pages 461 – 477. Springer Berlin Heidelberg.

[Lotte et al., 2007] Lotte, F., Congedo, M., Lécuyer, A., Lamarche, F., and Arnaldi, B. (2007). A review of classification algorithms for eeg-based brain-computer interfaces. *Journal of Neural Engineering*, 4(2):R1.

[Ludwig et al., 2009] Ludwig, K. A., Miriani, R. M., Langhals, N. B., Joseph, M. D., Anderson, D. J., and Kipke, D. R. (2009). Using a common average reference to improve cortical neuron recordings from microelectrode arrays. *Journal of Neurophysiology*, 101(3):1679–1689.

[Mallat and Zhang, 1993] Mallat, S. G. and Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415.

[Martinez and Escribano, 2008] Martinez, J. L. and Escribano, G. F. (2008). Distributed Video Coding. *Encyclopedia of Multimedia*, pages 194–198.

[MathWorks, 2017] MathWorks (2017). https://www.mathworks.com/.

[Mohamed and Justin, 2013] Mohamed, E. and Justin, D. (2013). From auditory and visual to immersive neurofeedback: Application to diagnosis of alzheirmer's disease. *Springer New York*.

[Oostenveld and Praamstra, 2001] Oostenveld, R. and Praamstra, P. (2001). The five percent electrode system for high-resolution eeg and erp measurements. *Clinical Neurophysiology*, 112(4):713 – 719.

[Ordonez et al., 2011] Ordonez, P., Armstrong, T., Oates, T., and Fackler, J. (2011). Using modified multivariate bag-of-words models to classify physiological data. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 534–539.

[Plinge et al., 2014] Plinge, A., Grzeszick, R., and Fink, G. A. (2014). A bag-of-features approach to acoustic event detection. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3704–3708.

[Pressel Coretto et al., 2017] Pressel Coretto, G. A., Gareis, I. E., and Rufiner, H. L. (2017). Open access database of eeg signals recorded during imagined speech. volume 10160, pages 1016002–1016002–11.

[Rennie et al., 2003] Rennie, J. D. M., Shih, L., Teevan, J., and Karger, D. R. (2003). Tackling the poor assumptions of naive bayes text classifiers. In *In Proceedings of the Twentieth International Conference on Machine Learning*, pages 616–623.

[Salama et al., 2014] Salama, M., Lashin, H., and Gamal, T. (2014). Recognition of unspoken words using electrode electroencephalograhic signals. *COGNITIVE 2014 : The Sixth International Conference on Advanced Cognitive Technologies and Applications*, pages 51–55.

[Selesnick et al., 2005] Selesnick, I. W., Baraniuk, R. G., and Kingsbury, N. C. (2005). The dual-tree complex wavelet transform. *IEEE signal processing magazine*, 22(6):123–151.

[Shih et al., 2012] Shih, J. J., Krusienski, D. J., and Wolpaw, J. R. (2012). Brain-Computer Interfaces in Medicine. *Mayo Clinic Proceedings*, 87(3):268–279.

[Simkin et al., 2014] Simkin, D. R., Thatcher, R. W., and Lubar, J. (2014). Quantitative eeg and neurofeedback in children and adolescents. *Child and Adolescent Psychiatric Clinics of North America*, 23(3):427 – 464. Alternative and Complementary Therapies for Children with Psychiatric Disorders, Part 2.

[Suppes et al., 1997] Suppes, P., Lin, L. Z., and Bing, H. (1997). Brain wave recognition of words. *Proceedings of the National Academy of Sciences*, 94(26):14965 – 14969.

[Technologies, 2012] Technologies, T. C. (2012). *10/20 System Positioning.*

[Toennies, 2012] Toennies, K. (2012). *Guide to Medical Image Analysis: Methods and Algorithms*. Advances in Computer Vision and Pattern Recognition. Springer London.

[Torres-García et al., 2013] Torres-García, A. A., Reyes-García, C. A., L., L. V.-P., and Ramirez, J. (2013). Analisis de señales electroencefalograficas para la clasificacion de habla imaginada. *Revista mexicana de ingeniería biomedica*, 34:23 – 39.

[Torres-García et al., 2012] Torres-García, A. A., Reyes-García, C. A., and Villaseñor-Pineda, L. (2012). Toward a silent speech interface based on unspoken speech. *Proceedings of biosignals*, pages 370 – 373.

[Torres-García et al., 2016] Torres-García, A. A., Reyes-García, C. A., Villaseñor-Pineda, L., and García-Aguilar, G. (2016). Implementing a fuzzy inference system in a multi-objective {EEG} channel selection model for imagined speech classification. *Expert Systems with Applications*, 59:1 – 12.

[Vaughan, 2003] Vaughan, T. M. (2003). Guest editorial brain-computer interface technology: a review of the second international meeting. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(2):94–109.

[Wang et al., 2013] Wang, J., Liu, P., She, M. F., Nahavandi, S., and Kouzani, A. (2013). Bag-of-words representation for biomedical time series classification. *Biomedical Signal Processing and Control*, 8(6):634 – 644.

[Wang et al., 2015] Wang, P., Lu, J., Zhang, B., and Tang, Z. (2015). A review on transfer learning for brain-computer interface classification. In *2015 5th International Conference on Information Science and Technology (ICIST)*, pages 315–322.

[Wikipedia, 2017] Wikipedia (2017). https://es.wikipedia.org/.

[Wolpaw et al., 2002] Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., and Vaughan, T. M. (2002). Brain computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767 – 791.

[Yeh et al., 2013] Yeh, C. C. M., Su, L., and Yang, Y. H. (2013). Dual-layer bag-of-frames model for music genre classification. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 246–250.

[Zarate, 2005] Zarate, L. E. M. (2005). Análisis visual del electroencefalograma. *Guía Neurologica 7.*

[Zhao and Rudzicz, 2015] Zhao, S. and Rudzicz, F. (2015). Classifying phonological categories in imagined and articulated speech. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 992–996.