UNIVERSITATEA "POLITEHNICA" DIN BUCUREȘTI

FACULTATEA DE ELECTRONICĂ, TELECOMUNICAȚII ȘI TEHNOLOGIA INFORMAȚIEI

TEZĂ DE DOCTORAT

CONTRIBUȚII LA DEZVOLTAREA METODELOR DE ANALIZĂ A SCENELOR AUDITIVE ÎN VEDEREA RECUNOAȘTERII VORBIRII CONTINUE

CONTRIBUTIONS TO COMPUTATIONAL AUDITORY SCENE ANALYSIS METHODS FOR CONTINUOUS SPEECH RECOGNITION

AUTOR:
Ing. Valentin Andrei

CONDUCĂTOR DE DOCTORAT:
Prof. Dr. Ing. Corneliu Burileanu

COMISIA DE DOCTORAT

PREȘEDINTE	Prof. Dr. Ing. Gheorghe Brezeanu	Universitatea "Politehnică" București
CONDUCĂTOR	Prof. Dr. Ing. Corneliu Burileanu	Universitatea "Politehnică" București
REFERENT	Prof. Dr. Ing. Daniela Tărniceriu	Universitatea Tehnică "Gh. Asachi" Iași
REFERENT	Prof. Dr. Ing. Corneliu Rusu	Universitatea Tehnică Cluj-Napoca
REFERENT	Prof. Dr. Ing. Constantin Paleologu	Universitatea "Politehnică" București

BUCUREȘTI
2015

ACKNOWLEDGEMENT

I am extremely grateful to my PhD study coordinator Prof. Dr. Eng. Corneliu Burileanu for his constant guidance, support and feedback during the entire research activity. He developed a highly skilled research department which provided me with all the needed support and had a crucial contribution to the quality of my work.

I want to thank Lect. Dr. Eng. Horia Cucu and Lect. Dr. Eng. Andi Buzo for their valuable support and suggestions in improving my research, the published papers and for helping me add better scientific relevance to my studies.

I also want to thank Prof. Dr. Eng. Constantin Paleologu for having an important role in my decision to start a PhD and to Lect. Dr. Eng. Eduard Popovici and Prof. Dr. Ing. Octavian Fratu for introducing me to the academic research domain during my license and master cycles.

My thanks go to the evaluation committee members: Prof. Dr. Eng. Daniela Tărniceriu, Prof. Dr. Eng. Corneliu Rusu and Prof. Dr. Eng. Constantin Paleologu for accepting to review my thesis and for their valuable comments and remarks.

My colleagues at Intel Romania helped me with a fraction of their free time for collecting all the data related to the perception studies presented in this thesis. I am deeply grateful for this.

My highschool mathematics professor, Prof. Corneliu Voicilaș taught me that any mathematical or engineering problem has a solution and I must not give up until I have a solid argument for each conclusion. This philosophy guided my academic and career path and I am deeply grateful to him. Thank you to my highschool friends from C.N. "Ion Luca Caragiale" from Ploiești. We are a great group!

I want to thank my parents who supported and ensured that I receive the best education. Thank you for that and I will always be grateful.

Finally my gratitude and my love go to my wife Cornelia who patently supported me during my studies. She was always by my side encouraging me to move forward and to finish my thesis. Thank you!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 THESIS MOTIVATION

We live in a world where speech analysis technology has made its entrance into the multibillion dollar market of mobile and – why not – wearable devices. Our phones can be controlled using spoken natural language and now even smart watches or glasses base their user interaction on speech. This is indeed encouraging considering that the holy grail of recognizing natural speech automatically by a machine has been studied for more than 6 decades.

In this context, the first question that rises is: "how close is science to being able to build a system that would understand speech as efficient as a human being?" To answer this let us try to understand the complexity of an apparatus able to understand and interpret spoken language automatically and the required training strategies in order to build such a system.

Let us try to translate in terms of amount of processing power and storage capacity, the brain's ability to understand speech. The human brain is thought to have around 115 billion neurons [Roth, 2005] and around 3.6×10^{14} synapses. There are multiple cerebral regions and processes involved in speech understanding and producing: Broca's area – responsible for speech production, Wernicke's area – responsible for speech acknowledging, memory cortex – responsible for speech understanding and information interpretation, selective auditory attention – capable of selecting the relevant stream of information and possibly unstudied regions that influence speech perception. Let us approximate that these crucial brain areas use only 5% of the entire synapses. That leads us to a neural network of approximately 1.8×10^{13} neural links. Simplifying each link to a 4 byte float number, we get a neural network of around 64 Terabytes. This is the estimated complexity of the apparatus that incorporates all elements necessary to understand and interpret information transmitted via speech. For this neural network to process a single second of speech (32KB/s using a 25ms time window) it performs $720 * 10^{12}$ floating point operations so the amount of needed computing power is 720 TFLOP – this is the capability of a small supercomputer.

Next step is to understand the size of the training set used to make the network function. Let us take as example an 18 year old individual and assume he listens and processes speech for 6 hours / day. Considering a sampling frequency of 16 kHz and 16 bits per sample, by the time he is 18, he had listened to more than 4250 Gigabytes of speech.

So, let us review our assumptions: in order to understand information transmitted through speech, a 64 Terabytes neural network, previously trained with 4.25 TB of data, has

to give an answer to incoming speech in real time. Currently modern large vocabulary continuous speech recognition systems (LV-CSR) are composed of pre-trained statistical elements like Hidden Markov Models (HMM), or Deep Belief Neural Networks (DBNN) backed up by a set of language modeling rules and can be stored in memory spaces only a few megabytes large.

So, based on our assumptions and on the current state and complexity of ASRS, we can firmly state that until obtaining a 99% accurate LV-CSR a long road of research activities lies ahead – according to the previous paragraph, it requires complexity increase of orders of magnitude. The author's personal opinion is that the progress made in speech recognition field will be closely coupled with technological advances towards demystifying the human brain's functioning. These advances have no other option but to be fueled by progress in the field of high performance computing.

So, in conclusion of this paragraph, to strengthen the thesis's motivation, speech recognition is a highly challenging research domain holding numerous "secrets" and with a great potential for expanding in the following years. We strongly believe that this work can add valuable contributions to this field and its resulting applications. While the language used for experiments in this thesis is Romanian, there is a high chance of applying the studied methods and algorithms to other languages without extensive tuning.

1.2 SPEECH ANALYSIS AND PROCESSING CHALLENGES

The scientific field of speech analysis and processing is responsible for a set of among the most challenging technical problems: determining and interpreting information transmitted through natural speech (generally mentioned as *speech recognition*) and identifying who spoke (generally mentioned as *voice or speaker recognition*). Another challenge in this field is producing intelligible speech with the help of a machine – referred to as *speech synthesis*. However, the latter is more related to general audio processing techniques and is not as coupled to artificial intelligence as speech or speaker recognition and, therefore, will not be studied in this thesis.

Due to the existence of multiple commercial applications that can exploit speech recognition, the topic is divided by the size of the operated vocabulary. So, we have short vocabulary (SV-SR: maximum hundreds of words), medium vocabulary (MV-SR: maximum 5000 words) and large vocabulary speech recognition (LV-SR: mostly accepted to operate on the entire vocabulary of a language). The robustness of a system in handling multiple speakers divides speech recognition (SR) systems in speaker dependent and speaker independent. Of course, the SRS always needs to address the particularities of each language so this results in another division category.

Speaker recognition is widely accepted as being an easier task than speech recognition and at this point it shows quite satisfying results. This sub-field of speech analysis and processing domain is also a sub-field of a scientific area called behavioral biometry. The topic is divided in speaker verification and identification. The two branches are sensibly different. First one assumes verifying through speech analysis that the declaration "*I am person X*" is correct. The latter means identifying the identity of the speaker from a set of possible choices. Voice recognition is split into phrase dependent and independent. Table 1.1 summarizes the different speech processing categories discussed in this paragraph.

Table 1.1 – Speech Analysis and Processing Branches

Speech Recognition		Speaker / Voice Recognition	
Criteria	**Category**	**Criteria**	**Category**
Vocabulary Size	Large Vocabulary – LV-SR Medium Vocabulary – MV-SR Small Vocabulary – SV-SR	Action Type	Speaker Verification Speaker Identification
Speaker Robustness	Speaker Dependent Speaker Independent	Phrase Robustness	Phrase Dependent Phrase Independent

In the following chapters we will focus mostly on speech recognition related issues but if it is related to the topic, statements that involve speaker recognition will be made.

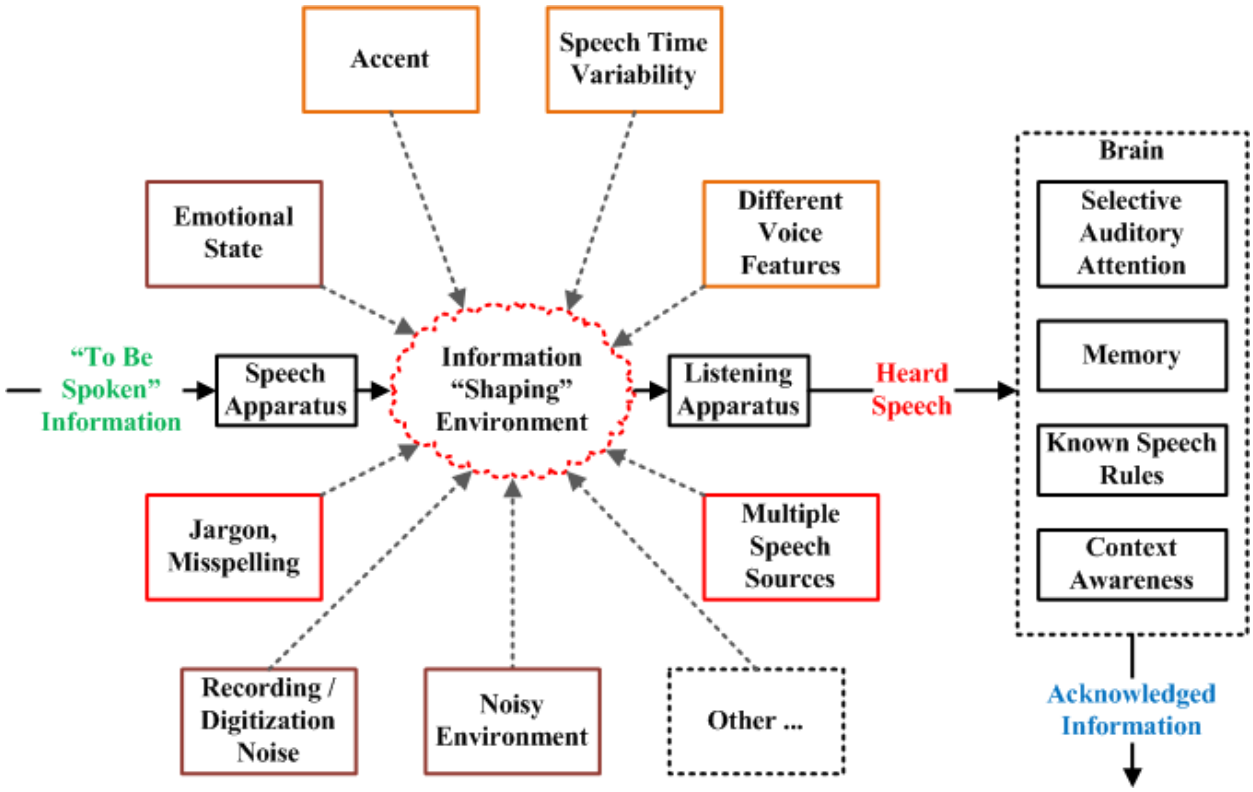

Figure 1.1 – Speech Recognition Challenges

Creating a LV-CSR assumes solving a large number of challenges. For example, the emotional state of an individual can greatly decrease the recognition ratio because if the subject is agitated he is likely to speed up his statements and have a different tone of voice, than when he is in a balanced state of mind. Figure 1.1 highlights several problems that need to be addressed when designing a LV-CSR.

Consider that Person A desires to transmit the information to Person B. The figure above proposes the fact that the speech encapsulating the information transmitted by Person A is being shaped or even worse, distorted by multiple factors. Person's B brain has to filter all the factors that are not relevant to the information, and reconstruct the acknowledged information. Next, let us discuss most important factors and their impact over the required complexity of a LV-CSR.

Recognizing speech in a multi speaker environment is a must for next generation LV-CSR but at this moment is one of the greatest obstacles that science needs to cross. Imagine

you want to dictate something to your Smartphone in a crowded call center where each person is engaged in another conversation that interferes with your speech. The brain performs selective auditory attention that solves this problem but until now, it has not been modeled by an algorithm that can be reproduced by a machine. It is widely accepted that by using both ears we can detect the location of the sound source but we are also able to focus to a relevant stream of spoken information even when we are presented just one channel recording with multiple speech sources. Currently using array of microphones to solve the problems brought by a multi speaker environment can be a solution but until selective auditory attention is not understood better, the challenges encountered in a multi speaker environment are likely to remain.

The speaker particularities are also an important factor to consider when designing a LV-SRS. It is not really a viable solution to provide speech recognition software that is not robust to different speakers. Currently, most of the application use training to adapt the mathematical models to the voice features of the user. However, if we meet a new person we are perfectly capable of understanding what he says without any adaptation. So, this additional training phase has to be viewed just as an intermediary step towards creating systems that are able to filter the voice features of each individual and extract only the information encapsulated by speech.

Accent robustness is a feature of a LV-SRS that is currently being widely studied. For example, an English SRS can work perfectly for a native English speaker but can be completely useless for an Indian user whose English speech is perfectly understood by another person.

One interesting question is related to the next factor we are bringing in discussion: an individual is perfectly capable of understanding the transmitted information even if the speaker has misspelled some words or is using jargon – can a LV-SRS be robust to jargon and misspelling? The human brain has an amazing error-correction mechanism that has to be better understood for increasing the possibilities of creating a 99% accurate LV-SRS.

Currently, there are software applications that can detect the "mood" of a person via speech. However, the emotional state can seriously degrade the word error rate (WER) of a LV-SRS. Labeling the mood is in essence an easier task because it involves mainly analyzing the voice's rhythm rather than analyzing the speech and the amount of information being searched is significantly smaller than in case of speech recognition.

In the past, there were other factors that affected speech recognition but currently they are not considered as main challenges. We have very accurate algorithms that can separate the moments of silence from speech moments, good filtering methods that can eliminate sounds not relevant for speech analysis. Also, the variability of speech in time is fairly well modeled with existing mathematical models.

At this point, natural language processing goes hand in hand with speech recognition. Actually, state of the art speech recognition needs language modeling. It is generally believed that in order to properly understand the spoken information, the brain uses information related to language like semantic and syntactic rules. For example, if we are in the middle of a conversation related to weather, it is unlikely to hear words related to biotechnology for example, and therefore we do not expect them. So, in order to acknowledge the heard speech the brain searches first only in a subset of the known words – the ones that are related to the conversation subject. Also, in a conversation we tend to anticipate what the person in front of us is about to say (mainly based on previous experiences) and it is easier to acknowledge the spoken information.

Natural language processing is a great partner for speech recognition also due to the potential commercial applications that can result from this symbiosis: real time speech translation. First application of this kind was realized by DARPA – [Voxtec, 2006] – and

involved translation of basic words from English to Arabic language, during Iraq and Afghanistan war.

Speech recognition currently does not need to use the entire knowledge related to machine translation but relies on a set of language models to verify the correctness of recognized speech from syntactic and semantic point of view.

1.3 PERFORMANCE METRICS

Defining stable performance metrics is critical for comparing different LV-CSR and to drive future research of improvement of the targeted systems. These metrics can focus on the accuracy of the recognition or towards the speed of the process. However, the latter is currently considered less important because the progress of the amount of available processing power is still within the law proposed by Moore. Table 1.2 provides an overview over most used performance metrics that describe LV-CSR.

Table 1.2 – Performance Metrics Describing Speech Recognition Systems

Performance Metrics	Description
Phone Error Rate (%) – **PER**	Being an error rate, the metric needs to be as low as possible and represents the ratio between the number of phones recognized incorrectly and the number of total phones.
Word Error Rate (%) – **WER**	Similarly, this metric represents the ratio between the number of whole words recognized wrong and the total number of words in the test speech.
Sentence Error Rate (%) – **SER** Phrase Error Rate (%) – **PhER**	Defines the ratio between the number of sentences that contain at least one word that was not properly recognized and the total number of sentences. As in a normal speech we have sentences and phrases combined, we will use a single metric to describe aggregated performance and due to notation simplicity we will choose SER.
Real Time Ratio – **xRT**	If this metric is 1, this means that the speed of speech decoding is the same as the rate of acquisition. Typically this ratio is lower than 1. The xRT is proportional with the complexity of the decoding system.

The metrics mentioned in the previous table are not the only existing but they are widely used to describe the performances of speech recognition systems. Generally, each study defines additional metrics, related to the subsection of the LV-CSR it targets.

1.4 ENGLISH SPEECH RECOGNITION

In this paragraph we will discuss about the current state of existing speech recognition applications designed for English language while the next paragraph will focus on Romanian language. Since this research domain started producing studies decades have passed and we can probably count in thousands the number of relevant results. Our opinion is that the deliverables of high quality studies were used by corporations in their products that tried implementing speech recognition. Some of the key stakeholders in this area are IBM, Google, Apple, Microsoft, Nuance and Carnegie Mellon University as an important player in open source community. Unfortunately, there are no detailed benchmarks or studies that asses the recognition ratio of publicly available LV-CSR.

Apple uses SIRI (claimed to be an intelligent personal assistant) as a key feature in marketing iPhone and iPad devices. While this strategy has probably increased the number of units sold, it also created a wave of complaints relatively to the feature's performance. For example, in [CNN, 2012] it is stated that SIRI is wrong in 38% of test cases. The experiment was made in an outdoor environment and was repeated in a quiet room resulting in a 6% reduction in error rate, to 32%. Test cases were basic web searches so we can label this error rate as SER. The study was made to compare Google's LV-CSR with SIRI and on the same test conditions it scored a 14% sentence error rate (SER).

Speech recognition was experimentally introduced in supervised data entry processes with minimal impact over decisions and one example is a system created to give simple answers to medical related questions transmitted via speech. In [Microsoft, 2011] SIRI and Nuance LV-CSR are benchmarked showing word error rates (WER) larger than 29% (achieved by SIRI). Nuance scored very poor for this test with only 67% WER. The test was made using a number of 120 medicine related phrases. It is interesting that even if the study was published on Microsoft's web page, it did not reveal results produced by MS Speech for the same test. Another important observation is the differences between WER of SIRI in the 2 test cases presented. This highlights the fact that benchmarking LV-CSR is very subjective and a standardized method of evaluating performance for such complex systems is yet to be defined.

IBM has more than 50 years tradition in speech recognition research, starting in 1962 with IBM Shoebox – being the most advanced system of its kind in that period. In 2011 an impressive demo was made using IBM Watson machine that competed in Jeopardy against top contestants and scored 3 times better than second place. The system used speech recognition and image recognition to understand the questions. Currently, the company delivers speech technology via IBM ViaVoice software. Unfortunately, studies targeting the performance of this software are not disclosed publicly to our knowledge.

Being an Open Source software pack, the Sphinx project maintained by Carnegie Mellon University, leaves the opportunity of efficient benchmarking due to the numerous regression tools [Marquard, 2009]. Table 1.3 highlights WER obtained for several dictated Yale courses.

Table 1.3 – CMU Sphinx4 Evaluation with Yale Courses

Course	Words	WER (%)
Bimolecular Engineering: Engineering of Immunity	6974	32%
Mating Systems and Parental Care	5795	40%
Personal identity, Part IV; What matters?	6603	49%
Maximilien Robespierre and the French Revolution	6643	61%

As we can see from the data above, the WER varies greatly depending on the complexity of the test speech and the size of the lexical field. While a history related text with complex phrases and a very large semantic domain shows 61% WER, a speech with complex terms but with relatively narrow semantic context can be easier to recognize by LV-CSR showing only 32%.

The conclusion that can be extracted from paragraph 1.4 is that, currently, English speech recognition software is yet unreliable and requires more time in correcting the errors than what is spared by reducing the need for key stroking for example. One important note is that English is by far the most studied and modeled language for speech recognition.

1.5 ROMANIAN SPEECH RECOGNITION

Romanian speech recognition is challenged by two additional facts, when compared to English. The first is represented by the lack of speech databases and language resources, essential for training complex mathematical models. The latter is a technical aspect and is related to the nature of Romanian, which is a morphologically rich language, which means that unlike English, it has a much larger vocabulary thus increasing the need for speech and language modeling resources.

Romanian speech recognition has more than 30 years tradition and started around 1980 within several research groups. In [Burileanu, 1983] one of the first isolated word recognition (IWR) systems is presented. The system was not speaker dependant. In [Furui, 1986], a similar system was disclosed, that used a dynamic time warping (DTW) approach to compare voice signals. At that time, processing power was very expensive and since the DTW algorithm has $O(n^2)$ complexity it was not considered a good approach for LV-CSR. In [Groza, 1984] one of the first attempts in recognizing vowels is being made. This could be considered as one of the first works towards LV-CSR in Romanian language though an almost 2 decade older similar work is presented in [Nicolau, 1963].

Starting with the 90's the computers became more widespread in the research environment and so on grew the more complex mathematical modeling capabilities. In [Grigore, 1998] one of the first studies that make use of neural networks in order to recognize vowels is presented. [Valsan, 1998b] presents a complex self organizing map (neural network) to quickly spot keywords in a spoken sequence.

After 2000, the works in the ASR field focused more and more on contributing to creating a LV-CSR for Romanian language. Reports of the first speech databases occurred in [Petrea, 2008]. Also, complex context modeling was presented in [Oancea, 2004]. The model was employed for a MV-CSR. One of the first Romanian works that raises the problem of speech "Understanding" is presented in [Militaru, 2009]. During the same period, automatic speech recognition studies were completed by linguistic studies like [Cristea, 2006]. One of the first complete LV-CSR for Romanian language is described in [Cucu, 2011]. In terms of

performance, [Cucu, 2011] reveals experiments made using CMU Sphinx that obtained WER as low as 22% for unknown speakers.

1.6 THESIS OBJECTIVES AND OUTLINE

In this thesis we selected one of the challenges discussed in the previous paragraph as focus: the ability of a person to fully understand speech in competing speakers' environments. However, the work is not mainly targeted towards separating speech sources, a field referred to as blind speech source separation. We also address this topic but we are more interested in contributing to computational methods for auditory scene analysis (CASAN). The methods and studies that we developed are analyzing speech because we see them as a step for contributing to a LV-CSR system.

The work touches also the area of speech perception. It is a fact that human listeners can follow multiple competing speakers but there is no knowledge of what is the maximum number of speakers that can be followed simultaneously. A LV-CSR system can be designed to approach the performances of a human listener but at the current stage there is small chance that the LV-CSR will work better in challenging conditions. This is why we designed an experiment that tries to determine how many competing speakers a person can follow. The experiment and its results are one of the novel contributions of this work.

In parallel, a set of methods for determining the speaker count in a recording with competing speakers were designed. The methods are also novel and we were able to compare their performance with what was achieved by human listeners. The algorithms can contribute to improving the accuracy of speech source separation, where the number of competing speakers is unknown. Basically, the competing speaker counting approaches enable an entire class of speech source separation methods, based on independent components analysis. The rest of the work is organized as follows:

- **Chapter 2** will highlight some of the current research trends in the area of speech recognition and understanding. We directed our attention mostly on "young" research fields encouraged as important topics at top international conferences but also covered some well established methods, widely used in this area. We will shift our presentation from phone recognition methods to emotion detection and jargon tolerance. While the chapter is a presentation of the most novel research in the domain, it also contains some personal opinions on future linked research areas like the usage of 3rd generation neural networks: spiking nets.

- **Chapter 3** is a deep dive analysis into voice activity detection techniques. We will highlight some of the widely adopted techniques for selecting speech but the focus will be on presenting methods that are adapted to function in multi-speaker environments.

- **Chapter 4** will be a thorough analysis of a novel set of methods for determining the number of competing speakers in a timeframe. We will go over a multitude of concepts ranging from studies that highlight the separation efficiency of various feature extraction methods to experimental results using statistical separation methods, neural networks and pattern alignment techniques. The performance of the speaker counting methods will be extensively discussed as the section contains an important part of the novel contributions brought by the thesis.

- **Chapter 5** will target blind speech source separation (BSPS). The methods discussed in section 4 enable independent component analysis (ICA) based BSPS and we present a synthesis of several ICA based approaches along with the achieved performances.

- **Chapter 6** will describe the perception experiments. This section is rather non-technical but it is extremely interesting to observe the limitations of human selective auditory attention. The first part will present the methodologies used (customized software, rules, speech sources, speech topics, etc.). The experiment was conducted in 2 phases with improvements added in the second phase. We will describe by comparison the 2 phases. The work is also one of the important contributions of this thesis.

- Finally, **chapter 7** will highlight the main conclusions of this work, putting emphasis on the novel concepts and findings that were added.

2 RESEARCH TOPICS IN SPEECH RECOGNITION

Speech recognition is a widely studied topic across the scientific community. Scientists are not only focused directly towards creating methods for recognizing continuous speech but also to a vast set of adjacent topics. The holy grail of this research area is to understand the biological aspects that enable humans to understand speech, completed by an in depth understanding of languages evolution. Once this is accomplished, researchers need to have mathematical instruments to create or at least approximate the phenomenon in an artificial environment. Computing power is also a major requirement for future speech recognition systems.

This chapter tries to give a brief image over the current research made in the field of speech recognition and correlated topics. A first overview can be given by the scientific areas covered by Interspeech 2015 conference, [Interspeech, 2015]. Firstly, the conference reviews work done in the *speech perception, production and acquisition field*. This includes models of speech production, the physiology and neurophysiology of speech production along with neural basis of speech production. Models of speech perception are also studied in correlation with the interaction between speech production and speech perception. Cognition and brain studies on speech are encouraged. The conference is also interested in multilingual studies in order to get a better understanding on the diversity of languages and specifics of each.

Papers related to *phonetics, phonology and prosody* are also being published every year, covering observations and studies of various languages, discourse and dialog structures, sound related changes, etc. *Speaker and language identification* plays an important part in the research topics related to speech. Language identification and verification, dialect and accent recognition, speaker verification and identification, robustness to variable and degraded channels, speaker confidence estimation, speaker diarization and multimedia speaker recognition are some of the topics studied by this area related to speech study. A relatively new topic, covered by mixed research teams including engineering researchers, medical researchers and psychologists is related to *analysis of paralinguistic in speech and language*. This covers analysis of speaker states and traits, pathological speech and language, non verbal communication, social and vocal signals, singing, perception of paralinguistic phenomena, etc.

The next topics covered by [Interspeech, 2015] fall within the engineering space. *Speech coding and enhancement* is a major topic, and covers speech coding and transmission, low-bit-rate speech coding, perceptual audio coding of speech signals, speech enhancement in single and multi channel, speech intelligibility, active noise control, de-reverberation for speech signals echo cancelling, etc. *Speech synthesis and spoken language generation* is an essential component towards creating a fully speaking machine. Grapheme to phoneme

conversion, text processing, signal processing and statistical models for speech synthesis are some of the interesting sub-topics. These are completed by research around articulatory speech synthesis, synthesis of singing voices, prosody modeling and generation, expression, emotion and personality generation, etc. Methods targeting speech recognition engines cover three major topics: *signal processing, acoustic modeling, robustness and adaptation* followed by *architecture, search and linguistic components* and completed by *technologies and systems for new applications*. These two major topics cover papers related to the mathematical methods used for speech analysis, phone recognition and evaluation. Also, methods of improving speech driven applications are of important interest. The latter two topics fall within natural language processing space: *dialog, summarization, understanding* completed by *translation information retrieval and resources*.

Following paragraphs will cover some of the widely studied topics in speech recognition. The first section will highlight how a specific language can influence the complexity of a speech recognition system. Phone recognition will be treated in the next sub-chapter. We will focus on feature extraction methods like MFCC and PLP along with widely used methods for feature classification: HMM-GMM or DNN based methods. In the same chapter we will also bring in discussion the usage of spiking neural networks. Language modeling always plays an important part in LV-CSR systems and it will be discussed in chapter 2.4. Then we will continue with describing some of the existing speech databases that can be used for training and testing the algorithms. We will also bring into discussion methods related to blind speech source separation but this will be treated in more detail in chapter 4. More "exotic" topics like accent, emotion filtering or jargon will briefly be described in chapters 2.6 and 2.7.

The mentioned sub-chapters can also be "mapped" to the components proposed in figure 2.1. A diagram of a LV-CSR system is proposed. It contains functional blocks related to speaker adaptation, phone recognition aided by voice signal compression, accent and emotion filtering stages along with language models where jargon is also being considered.

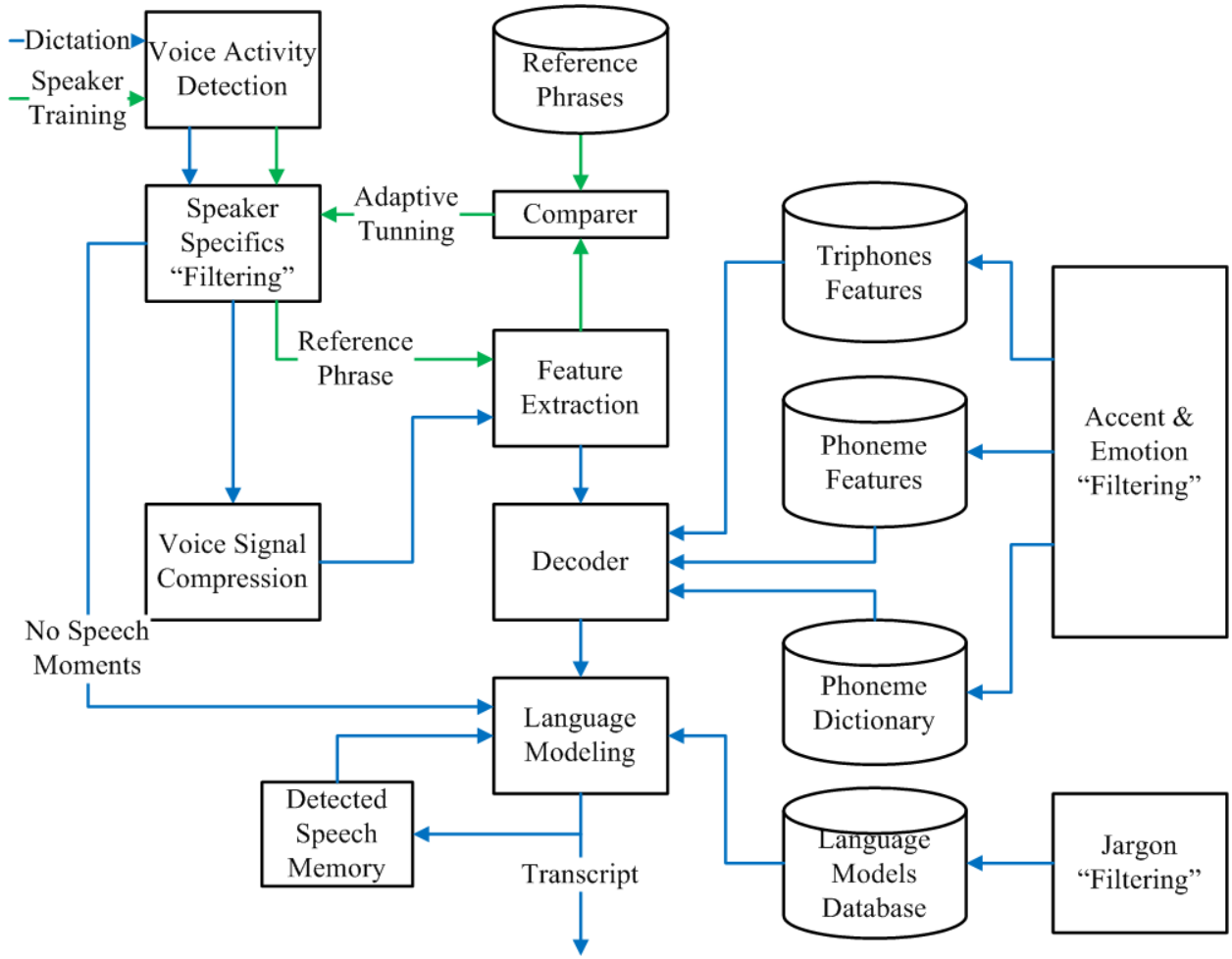

Figure 2.1 – Architecture of a LV-CSR

2.1 INFLUENCE OF SPECIFIC LANGUAGES

A specific language can influence one theoretical linked speech recognition and understanding engine in multiple ways. Vocabulary size is the first parameter of a language reflected on when analyzing the feasibility of creating a speech recognizer. Although it is difficult to determine the size of a vocabulary because there is no fair method of telling what to count, some languages due to their history and specifics are more likely to develop a higher number of words. Of course, this is closely coupled to the technological state of the culture that speaks that language. However, vocabulary size is an important factor because it divides speech recognition driven applications into three (or perhaps four) categories:

- Small vocabulary speech recognition – the applications are targeted to recognizing only a small set of words. The methods can be used for recognizing spoken commands, for example, for maneuvering a machine and being able to command non critical functionalities without a button. These applications have the great advantage that they are more robust to speaker variability.

- Medium vocabulary speech recognition – targeted for recognizing a set of words related to a specific topic of discussion. Methods in this space are used for example in call center robots or speech driven question and answer systems that aim to

automate some redundant tasks. Also, speaker robustness is relatively high for such applications.

- Large vocabulary speech recognition – targeted for recognizing continuous speech. While speaker robustness is not that high, it is the type of application with highest usage possibilities. Automatic dictation is the most well known but the application spectra is widened towards advanced human machine interaction. LV-CSR systems are starting to be developed towards unlimited vocabulary continuous speech recognition, which can be abbreviated UV-CSR.

The number of phonemes in a specific language is also an important parameter to be taken into consideration. There are great differences between the numbers of phonemes in different languages. Studies reveal a trend of decreasing phoneme count for a specific language correlated with the distance from southern Africa. For example, [Pereltsvaig, 2014] reveals that as certain languages are being spoken farther from southern Africa, the number of phonemes in that language decreases. This is closely correlated with human migration path. For example, there are languages in Africa, called "click languages" with as far as 141 phonemes. German language has 41 phonemes, Mandarin language has 32 phonemes, Garawa language spoken in Australia has 22 phonemes, Hawaian languages have 13 phonemes while the South American Piraha language has only 11 phonemes. Piraha is a language spoken by a tribe of hunter-gatherers in the Amazon basin. Unfortunately, these exotic languages are spoken only in isolated communities and due to this fact, creating a database to be used for training and testing a speech recognition engine is a quasi impossible task. Although there is no rule, it appears that Piraha's language lexicon is small. For example, in [Brockman, 2007], the Piraha language is described as having no words for numbers.

The complexity of a language's grammar is also a key factor to consider when creating a LV-CSR system. According to [Perfors, 2010], the human language has the capacity of generating a potential infinite number of possible sentences. This capacity can result from an underlying generative mechanism – called grammar. The complexity of a grammar generally makes a certain language difficult to learn. This can translate in the difficulty of creating an algorithm or a system that can function similarly to the grammar. The number of phonemes in a language is not directly linked with the complexity of the grammar. For example, Turkish language is heavily agglutinating so that a single word can mean a complex sentence. This makes the grammar rules very complex. However, Turkish has a medium number of phonemes but the Tuyuca language of Amazon is even more agglutinating with a very small number of phonemes.

Although the goal of this work is not related to studying language related specifics, it is important to conclude that language plays a key part in shaping a LV-CSR. The topic is studied widely across the scientific community.

2.2 PHONE RECOGNITION

Phone recognition refers to being able to produce a succession of phones using an input speech signal. It is generally accepted that the speech signal is not fed directly into a Hidden Markov Model (HMM) – Gaussian Mixture Model (GMM) or neural network based processing chain. There is an additional step, usually called feature extraction. A high percentage of LV-CSR applications use small variations of the following feature extraction methods:

- Perceptual Linear Prediction (PLP)
- Mel-Frequency Cepstral Coefficients (MFCC)

In [Honig, 2005] we can review each of the methods. The paper states that PLP method is more robust when there is an acoustic mismatch between training and testing data but under clean conditions and where there is no significant mismatch, MFCC approach can yield slightly better results. However, the two methods are close in terms of performance.

2.2.1 MFCC parameterization

Let us briefly describe the MFCC approach, as documented in [Davis, 1980]. We are interested in computing computing the MFCC, $c_{i,j}$, and also the derived Dynamic Coefficients (i.e., Delta Coefficients, $\Delta_{i,j}$, and Delta Delta Coefficients, $\Delta^2_{i,j}$) for a given input signal. The algorithm's main steps are described in the following. Let us denote by N the number of MFCC generated for each window.

First, the input speech signal is split into M overlapping windows with a fixed number of samples L: $w_1(n)$, $w_2(n)$, ..., $w_M(n)$. Usually the overlapping fraction varies between 0.25 and 0.75, and the window's time length from 10 milliseconds to 50 milliseconds. A window can be multiplied by a weighting function like Hamming, Triangular, etc. Then, we proceed to computing the set of N coefficients for all the signal's windows, as described by the following steps.

For a k indexed window, the Fast Fourier Transform (FFT) is computed generating the short time spectrum $S_{\mathrm{Wk}}(f)$. The window's spectrum is then passed through each FFT filter from the bank illustrated in figure 2, resulting a set of filtered spectrums: $S1_{\mathrm{Wk}}(f)$, $S2_{\mathrm{Wk}}(f)$, ..., $SN_{\mathrm{Wk}}(f)$. The decimal logarithm of the spectral power generated by each filtering operation is computed according to (2.1), and inserted into a vector P with N elements, i.e.,

$$P(i) = \log_{10}\left[\sum_{\mathrm{f}=0}^{N_{FFT}} Si_{\mathrm{Wk}}(f)\, Si^{*}{}_{\mathrm{Wk}}(f)\right] \tag{2.1}$$

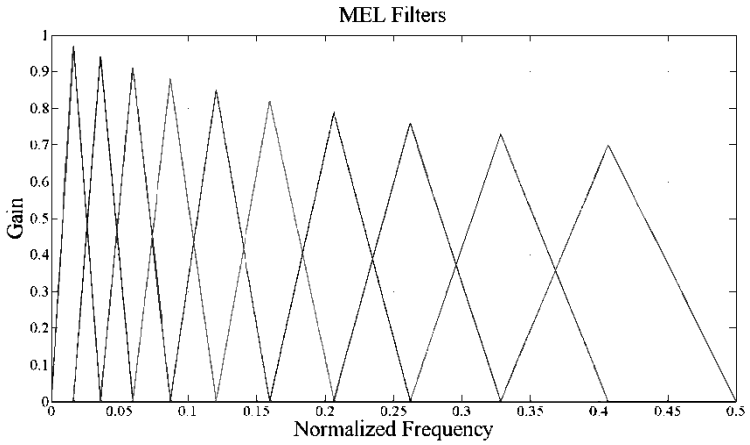

Figure 2.2 – Mel-Frequency Filter Banks

A Discrete Cosine Transform (DCT) is applied to the constructed vector, according to (2.2), generating the set of MFCC for the selected window k:

$$C(i,k)=\begin{cases}\dfrac{2}{\sqrt{N}}\sum\limits_{\mathrm{q}=0}^{N-1}P_{\mathrm{q}}\cos\left[\dfrac{\pi}{\mathrm{q}}\left(\mathrm{q}+\dfrac{1}{2}\right)i\right] & \text{if } i\neq 0\\ \dfrac{1}{\sqrt{2}}\sum\limits_{\mathrm{q}=0}^{N-1}P_{\mathrm{q}} & \text{if } i=0\end{cases} \tag{2.2}$$

In the end, the resulting MFCC will be stored in a matrix, as follows:

$$C=\begin{vmatrix} c_{1,1}c_{1,2}.....c_{1,M}\\ c_{2,1}c_{2,2}.....c_{2,M}\\ .....\\ c_{N,1}c_{N,2}.....c_{N,M}\end{vmatrix} \tag{2.3}$$

Also, we can add another coefficient, $c_{\mathrm{i},N+1}$, to the obtained matrix by computing the signal window's energy as

$$c_{N+1,\mathrm{k}}=\sum_{n=0}^{L}\left|w_{\mathrm{k}}(n)\right|^2 \tag{2.4}$$

The above coefficients matrix will be used in the following steps to compute the derivate Dynamic Coefficients ($\Delta_{i,j}$ and $\Delta^2_{i,j}$), according to.

$$\begin{aligned}\Delta(p,q)&=\frac{\sum\limits_{i=1}^{k}i(C'[p+i,q]-C'[p-i,q])}{2\sum\limits_{i=1}^{k}i^2}\\ \Delta^2(p,q)&=\frac{\sum\limits_{i=1}^{k}i(\Delta'[p+i,q]-\Delta'[p-i,q])}{2\sum\limits_{i=1}^{k}i^2}\end{aligned} \tag{2.5}$$

The method described above is being used by Matlab Voicebox functions [Brookes, 2002]. The equations are very similar to the ones describing a finite impulse response filter, except that the output is delayed with a number of *k* cycles. We can use this approach to determine the Delta Coefficients by following the steps below. To compute the Delta Delta coefficients, the process is identical, except that the *k* parameter can be different. We copy the first line of *C* matrix on top of it, by a number of *k* times. We copy at the bottom of *C* matrix, its last line by a number of *k* times. In order to obtain the Delta Coefficients as a column in a Δ matrix, we apply $h(n)$ defined as in (2.6) to the corresponding column in C' (the newly modified *C* matrix).

$$h[n]=\frac{-(n-N)}{2\sum\limits_{i=1}^{N}i^2} \tag{2.6}$$

In order to obtain the first order Delta Coefficients, we need to apply the algorithm described above, having the *C* matrix as input. Then, we will consider the resulting Δ matrix

as input to the same algorithm (eventually after modifying the *k* parameter) with the purpose of computing the Δ^2 matrix.

After completing all the steps described in the current sub-section, the input speech signal will be transformed into a set of values representing the description of the speaker's voice characteristic. The values will be stored into an *Y* matrix, defined as

$$Y = \begin{pmatrix} C \\ \Delta \\ \Delta^2 \end{pmatrix} \tag{2.7}$$

The matrix *Y* will be considered the *feature matrix*. Researchers use MFCC coefficients due to 3 main reasons: the "Melody" perceptual scale provides the best modeling of the human hearing system which perceives much better the frequencies below 1000 Hz – therefore the used filter banks provide softer attenuation in this frequency range. Another argument would be that the MFCC coefficients are not only effective for speech recognition but also for speaker recognition / identification. Furthermore, MFCC coefficients produces a set of uncorrelated thus containing more information about the underlying process and helping the classifier stages – as described below: HMM-GMM, DNN, etc.

2.2.2 PLP parameterization

[Hermansky, 1990] states that PLP method uses 3 psycho-acoustic concepts for modeling the spectra perceived by human hearing system. These concepts are the critical band resolution, equal loudness curves and the sound intensity variation. Figure 2.3 highlights the steps for computing the PLP features of a given sound signal.

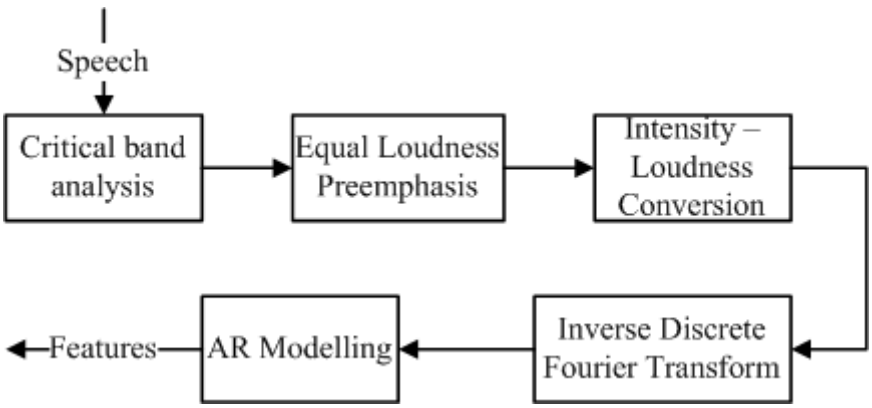

Figure 2.3 – PLP features computation

Before the critical band analysis step, the speech signal is divided into Hamming windows using (2.8). The usual window length is 20 milliseconds. After the computation of windows, a FFT transform is applied and then the signal's power spectrum is computed using (2.9).

$$W(n) = 0.54 + 0.46\cos[2\pi n/(N-1)] \tag{2.8}$$

$$P(\omega) = \mathrm{Re}[S(\omega)]^2 + \mathrm{Im}[S(\omega)]^2 \tag{2.9}$$

The critical band analysis transposes the power spectrum into a new frequency scale called "Bark scale" described in [Havelock, 2008] and given by (2.10). A convolution between the translated power spectrum and a simulated critical band spectrum is computed. The simulated spectrum is given by (2.11) while the convolution is being given by (2.12).

$$\Omega(\omega) = 6\ln[\omega/1200\pi + \sqrt{(\omega/1200\pi)^2 + 1}] \tag{2.10}$$

$$\psi(\Omega) = \begin{cases} 0 \text{ for } \Omega < -1.3 \\ 10^{2.5(\Omega+0.5)} \text{ for } -1.3 \le \Omega \le -0.5 \\ 1 \text{ for } -0.5 < \Omega < 0.5 \\ \dfrac{1}{10^{(\Omega-0.5)}} \text{ for } 0.5 \le \Omega \le \sim 2.5 \\ 0 \text{ for } \Omega > 2.5 \end{cases} \tag{2.11}$$

$$\theta(\Omega_i) = \sum_{\Omega=-1.3}^{2.5} P(\Omega - \Omega_i)\Psi(\Omega) \tag{2.12}$$

The next step is multiplying the spectrum given in (2.12) with an expression simulating the "equal loudness" curve. This represents the uneven sensibility of the human hearing system to different frequencies. The expression of the "equal loudness" curve is shown in (2.13) as referred in [Hermansy, 1990]. (2.13) function is applied to a given signal using (2.14).

$$E(\omega) = [(\omega^2 + 56.8\times10^6)\omega^4]/[(\omega^2 + 6.3\times10^6)^2 \times (\omega^2 + 0.38\times10^9)] \tag{2.13}$$

$$E[\Omega(\omega)] = E(\omega)\Theta[\Omega(\omega)] \tag{2.14}$$

Before the PLP coefficients are to be estimated the signal has to be compressed using an autoregressive model, given in (2.15), then by using the Yule-Walker equations as described in [Hermansky, 1990] and given by (2.16) or (2.17) the PLP features can be derived.

$$\phi(\Omega) = E(\Omega)^{0.33} \tag{2.15}$$

$$r_{xx}(k) = \begin{cases} -\sum_{l=1}^{N} a_l r_{xx}(k-l) + \sigma^2; k = 0 \\ -\sum_{l=1}^{N} a_l r_{xx}(k-l); k \ge 1 \end{cases} \tag{2.16}$$

$$\begin{bmatrix} r_{xx}(0)r_{xx}(-1).....r_{xx}(-(N-1)) \\ r_{xx}(1)r_{xx}(0).....r_{xx}(-(N-2)) \\ ..... \\ r_{xx}(N-1)r_{xx}(N-2).....r_{xx}(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ ... \\ a_N \end{bmatrix} = \begin{bmatrix} r_{xx}(1) \\ r_{xx}(2) \\ ..... \\ r_{xx}(N) \end{bmatrix} \tag{2.17}$$

$$\sigma^2 = r_{xx}(0) + \sum_{l=1}^{N} a_l r_{xx}(-l)$$

Using the above algorithms, for a given signal window, a set of N PLP features will be generated. In the above expressions, the PLP coefficients are a_1, a_2, ... a_N. The PLP feature computation needs to be served with the following input parameters:

- The sampling frequency and the number of bits per sample
- The length of the analysis window
- The type of the analysis window
- The number of FFT points
- The expression of the equal loudness curve
- The order of the autoregressive model

In the above paragraph we presented the PLP features computation algorithm as it was described by the author in [Hermansky, 1990]. From then on a number of variations have been developed, mainly focused on the optimal parameters selection: [Shrawankar, 2010], [Alam, 2012], [Mporas, 2007].

2.2.3 HMM-GMM methods

According to [Ghrahramani, 2001] Hidden Markov models (HMM) are an excellent tool for modeling of time series data. The first paper describing HMM methods is [Baum, 1966]. However, applications of the methods have appeared slightly later. They are used extensively in speech recognition systems but they are not limited to this field. For example, molecular biology, data compression and multiple areas related to pattern recognition are using HMM based methods. Computer vision algorithms also use HMM for example in sequence modeling and object tracking. In speech recognition HMM methods are being used for more than two decades. In [Rabiner, 1989] and [Rabiner, 1993] we can read some of the first applications of HMM in speech recognition.

In a Markov process all the following states do not depend of the previous states. A hidden Markov model is a statistical Markov model where the process that is under modeling is assumed to be a Markov process with hidden (unobserved) states. Each state has a certain distribution probability depending on the output observations. In consequence the sequence of observed output states give some information about the underlying succession of input states. A Markov process can be described by (2.18) and if it is characterized by discrete states it can be described by (2.19).

$$\begin{aligned} &...p_n < p_{n-1} < ... < p_2 < p_1 < s < t... \\ &P\{X(t) = x(t) \Big| \begin{matrix} X(s) = x(s), X(p_1) = x(p_1), X(p_2) = x(p_2),...\} \\ = P\{X(t) = x(t) \big| X(s) = x(s)\} \end{matrix} \end{aligned} \tag{2.18}$$

$$P\{X_n = x_n \,|\, X_{n-1} = x_{n-1}, ..., X_0 = x_0\} = P\{X_n = x_n \,|\, X_{n-1} = x_{n-1}\} \tag{2.19}$$

A typical architecture of a HMM is presented in figure 2.4. The *x1, x2 and x3* are the sequence of states; *y1, y2, y3 and y4* are the possible observations while the *a12, a21 and a23* are transition probabilities.

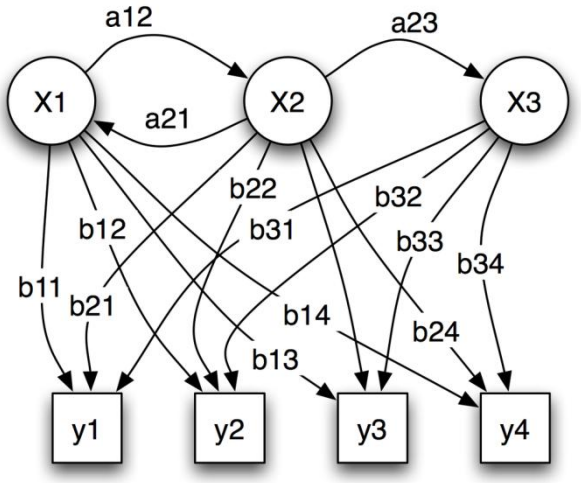

Figure 2.4 – Typical architecture of a Hidden Markov Model

The transition probability tells the probability of the system switching from a given state to another. This metric is defined by (2.20). Because the transition probability does not depend on time, the values need to satisfy the (2.21) relation. The main idea of the algorithm, as described in [Rabiner, 1993] is to determine a set of *X = x(1), x(2), …, x(T)* states that have the highest probability of generating the sequence of *Y = y(1), y(2), …, Y(T)* observations. This is described by (2.22).

$$a_{ij} = \mathrm{P}(x(t+1) = j \,|\, x(t) = i) \tag{2.20}$$

$$\sum_{j=2}^{N} a_{ij} = 1 \tag{2.21}$$

$$P(X_{1:T}, Y_{1:T}) = P(X_1)P(Y_1 \mid X_1)\prod_{i=2}^{T} P(X_i \mid X_{i-1})P(Y_i \mid X_i) \tag{2.22}$$

In order to determine the probability described by (2.22) a transition matrix $P(X_i|X_{i-1})$. This is generally done using a training phase. Considering figure 2.1, the HMM methods are used in the decoding stage of the LV-CSR processing chain. The following paragraphs will reflect the specifics of using HMM in speech recognition systems, as iterated by [Gales, 2007].

First, consider that each word is a sequence of base phones, denoted K_w. This is called pronunciation of the word and can be written as $q_{1:Kw}=q_1, \ldots, q_{Kw}$. The set of observations involved in the Markov process are the elements of *Y* matrix defined in (2.7) – the actual set of MFCC coefficients. Because each word can have different pronunciations the likelihood that a pronunciation can lead to a set of observations associated to a word *w*, must be

computed over multiple pronunciations. We denote as *Q* the sequence of all particular pronunciations of word *w*. However, an analysis over the elements in the *Q* set will reflect a rather low degree of similarity mainly because each pronunciation depends on the speech context. If HMM is used considering single phones (often referred as *monophones* in [Gales, 2007]) there is a high chance that the system will not be robust to natural speech where each word is spoken differently depending on the context. This problem is being solved in LV-CSR systems by using a unique phone model for every possible pair of left and right neighbors. The models are called *tri-phones*. If a specific language holds N phones, there is a N^3 sized set of possible tri-phones.

Generally, HMM are used in conjunction with Gaussian mixture models (GMM). According to [Reynolds, 2001], a GMM is a parametric probability density function, represented as a weighted sum of Gaussian component densities. GMM are often used as parametric models of the probability distribution of continuous measurements of features in a biometric system such as vocal-tract related spectral features in a speaker recognition system. A training data is necessary to estimate the parameters of a GMM. Several algorithms can be used for that, like Expectation-Maximization (EM) or Maximum A Posteriori (MAP) also defined in [Reynolds, 2001].

2.2.4 Deep neural networks

Deep learning is starting to be used more often in pattern recognition challenges and especially in speech processing related applications. The traditional methods like MFCC, PLP, HMM and GMM have a long history of development but have evolved also considering processing power limitations. Fortunately, Moore's law is true for more than 50 years after its phrasing so nowadays computing power is not necessarily an issue. However, it is obvious that in speech recognition applications the response has to be in real time and therefore the computational requirements of the method need to be reasonable but with the evolution of processors the balance between processing power requests and system precision turns in favor of precision. Therefore, deep learning has gain ground. Deep learning architectures are not necessarily extremely different than artificial neural network architectures but have some specifics. While artificial neural networks have been around for more than 4 decades, the reference papers describing the usage of deep neural networks in speech recognition systems are [Hinton, 2010], [Mohamed, 2010], [Dahl, 2012] and [Hinton, 2012]. In [Hinton, 2012] the authors state that due to the advances in both machine learning techniques and also processing power, methods used for training deep neural networks (DNN) have become more efficient. This makes it possible that DNN can replace GMM in the phone decoder component of a LV-CSR processing chain. The referenced paper is a common view of some of the key players in the field of speech recognition: Google, IBM Research, Microsoft Research and University of Toronto.

A DNN is a classic feed-forward Artificial Neural Network (ANN) that has more than one layer of hidden units. Because of that, DNN can be trained using ANN specific training algorithms like variations of the classical Back-Propagation algorithm (BP). While DNN are able to learn sets of unsupervised data, they can also be fine tuned using supervised learning. As described in [Hinton, 2012], the DNN can be easily linked to a HMM because it can output the probabilities in the form of *p(HMM State | Acoustic Input)*. To demonstrate the efficiency of DNN, the authors showcase in [Hinton, 2012] a table with the Phone Error Rate (PER) metric for various combinations of system parameters, with experiments being done on TIMIT database. This is also shown in table 2.1.

Table 2.1 – Phone error rate for experiments done on TIMIT

Phone recognition method	PER (%)
CD-HMM in [Hifny, 2009]	27.3
Augmented Conditional Random Fields [Hifny, 2009]	26.6
Randomly Initialized Recurrent Neural Networks [Robinson, 1994]	26.1
Bayesian Tri-phone GMM-HMM [Ming, 1998]	25.6
Monophone HTMS [Deng, 2007]	24.8
Heterogeneous Classifiers [Halberstadt, 1998]	24.4
Monophne randomly initialized DNN (6 Layers) [Mohamed, 2012]	23.4
Monophone DBN-DNN (6 Layers) [Mohamed, 2012]	22.4
Monophone DBN-DNN with NMI training [Mohamed, 2012]	22.1
Tri-phone GMM-HMM W/ BMMI [Sainath, 2011]	21.7
Monophone DBN-DNNs on FBank (8 Layers) [Mohamed, 2012]	20.7
Monophone MCRBM-DBN-DNN on FBank (5 Layers) [Dahl, 2010]	20.5
Monophone Convolutional DNN on FBank (3 Layers) [Hamid, 2012]	20.0

So, the table above highlights how DNN methods have the chance of yielding lower PER. Due to this reason DNN was used in several commercial applications like: Bing, Switchboard, Google voice input, Youtube and English broadcast. In the above systems DNN were used as a replacement for GMM in the decoding processing stage. However, there are other possibilities of making use of DNN, for example, as to use a DBN-DNN in order to provide input features for GMM-HMM system. This has the disadvantage of requiring more computational power but may also combine the advantages related to accuracy of the two methods. Another approach is to use DNN in order to estimate articulatory features for detection based speech recognition. This approach, according to [Yu, 2012] achieved half the error rate of shallow neural networks with a single hidden layer.

2.2.5 Spiking neural networks

Looking at the results in table 2.1 we can draw the conclusion that the phone error rate is rather high and even more, the lower than 20% barrier does not appear to be crossed. Considering that the PER is displayed, the word error rate might be increased in a LV-CSR. Also, the TIMIT database is widely known and clean, thus it is unlikely to contain a large number of "corner cases". The study in table 2.1 would have been more interesting being done on a continuous speech database but since the results were reported in a 20 year interval we realize that this is a quasi impossible challenge.

Looking at HMM and DNN we can see that they are both statistical methods. Even though DNN are an evolution of neural networks, they are far from being able to simulate the behavior of even the simplest neural network in the human brain. The link between 2 neurons in a human brain, called synapse is by itself an extremely complex chemical process. In an ANN or DNN, the link between 2 neurons is usually given by a weight (a floating point number) and a predefined transfer function. The author's opinion is that the complexity of a synapse needs to be somehow modeled in software or hardware so that a system complex enough to understand spoken information is to be possible.

Spiking neural networks (SNN) are a step forward towards that direction. A brief overview of SNN can be read in [Moisy, 2009] and [Vreeken, 2003]. SNN are often referred to as the 3rd generation neural networks and they are mainly inspired from the brain's natural computing. Development of SNN is strongly aided by recent advances in neurosciences and according to [Moisy, 2009] they derive their strength from a reasonably fair representation of a real synapse. Although SNN research is rather new and heavily challenged by computational requirements, several researchers have made a goal out of using a SNN for pattern recognition tasks, especially when speech is involved. For example, [Verstraeten, 2005] presents a SNN based solution for isolated word recognition. While traditional methods have achieved good results for this specific task, the lecture is interesting because it presents a whole new approach with potentially good chances of evolution beyond the limits imposed by classical approaches. [Wang, 2005] are using SNN to represent auditory signals based on the properties of the spiking neuron refractory period. Also, speech recognition applications of SNN are explored in [Loiselle, 2005].

While the SNN related methods are perceived as being rather "exotic" and have not gained much traction in the speech recognition community, they are a new instrument worth being considered, mainly due their direct correlation between complexity and accuracy with advances made in computing power. While statistical methods are rather limited from the evolutionary perspective, because we often see that increasing the system's complexity (e.g. neurons) does not necessary produce better results, the performances of SNN methods are directly coupled with their size and complexity.

2.3 LANGUAGE MODELING

In day to day talk some phrases have an extremely similar sound to others. For example, in English, the sequences of words: "it is fun to recognize speech" and "it is fun to wreck a nice beach" can be easily taken one as another. The phone recognition algorithms can also easily generate incorrect words that sound alike. Language modeling is an essential step in recognizing speech that minimizes these types of errors. In other words, language modeling is useful for making the phone decoder to generate valid words, to contribute to generation of valid phrases and why not, to make it easier for the speech recognition engine to produce correct results. Language modeling techniques range from using plain simple dictionaries to implementing complex grammars. Considering that one of the most important applications of speech recognition is to create a system that does automatic translation of spoken information then we can say that speech recognition and natural language processing are in complete symbiosis.

One of the most important concepts in language modeling for speech recognition is the concept of *n-gram*. One of the first papers that use this concept is [Kuhn, 1990] which discusses several Markov language models. An n-gram is simply a contiguous sequence of n items from a given sequence of text or speech. Items can be phonemes, syllables, letters, words or base pairs according to the application. N-grams are generally collected from a text or speech corpus. Some particular cases of n-grams are unigram, bigrams and trigrams for successions of one, two respectively three items. A model based on n-grams tries to predict the N^{th} item considering the previous (N-1) elements. When referring to speech recognition, an n-gram model can be used as a complementary measure of generating a correct word. A simple diagram depicting a speech recognition engine using a simple n-gram model is represented in figure 2.5.

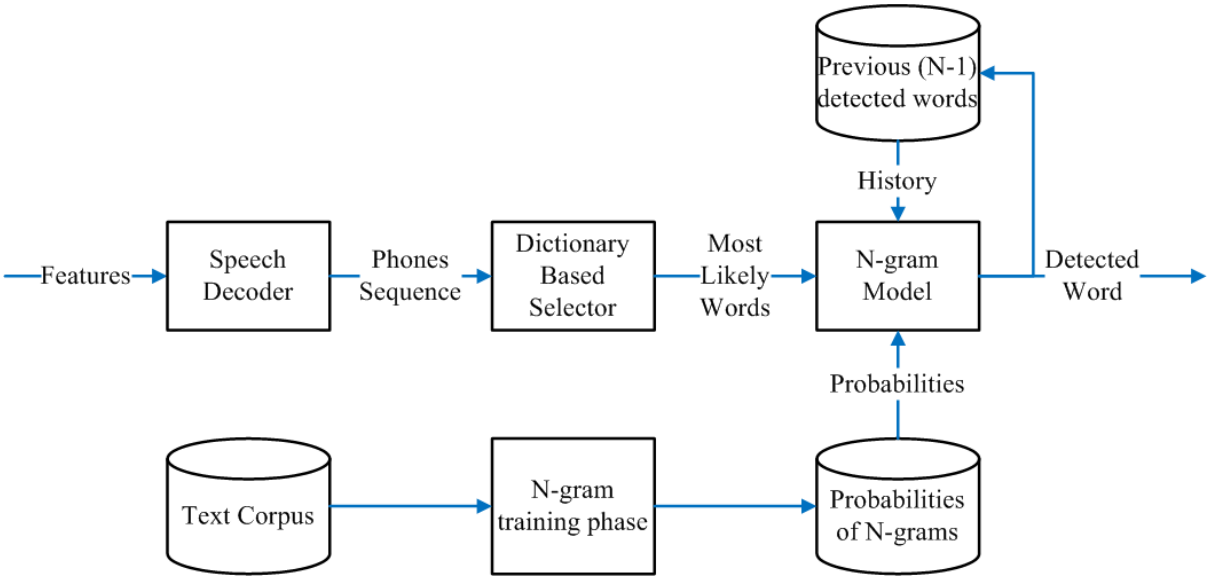

Figure 2.5 – N-gram model integration

Considering the above schematic, the n-gram is used to increase probability that the generated word is correct. For example, in [Lee, 1988] without a language model, the word error rate (WER) of a speaker independent, continuous-speech corpus using a 997 word vocabulary, was 29.4%. When integrating an n-gram model the WER dropped close to 4%. As stated before, the n-gram uses a history of N-1 words to predict the next word. The probabilities are based on frequencies and counts and can be easily integrated in the Viterbi search of a typical speech recognizer. Recently, Microsoft and Google started providing n-gram services for various research and development purposes.

A language model can contribute even further to a correct generation of word sequences. For example, based on a language model, a discussion topic can be derived. In simpler words, if the speech recognition engine detects with high frequency a set of words belonging to the quantum physics domain, it will be inclined to search related words within the same domain. In cases where it has to decide between words with similar probabilities of occurrence but with one belonging to the quantum physics topic and the other one to politics, knowing the topic of the discussion would improve the correct detection.

Language models do not necessarily need to be built based on whole words. For example, in the case of morphologically rich languages that do not have a high amount of pre-computed resources and statistics and suffer from "out of vocabulary" words, morphemes can be used. A morpheme is a sub-word and can be an excellent choice when the targeted language is not widely studied. For example, [Tachbelle, 2010] describes a morpheme based language modeling approach that also captures word-level dependencies. The study is targeted towards Amharic African language. There are also other studies having the same goal, for example [Choueiter, 2006] targets Arabic language and reveals how a 2.4% improvement in WER is achieved by using morphemes. [Saraswathi, 2007] targets Tamil language and demonstrates that a morpheme based language provides lower WER than a traditional bigram or trigram language model. In [Mousa, 2011] a morpheme based language model is tried for German due to the fact that German is a highly inflectional language and a large number of words can be generated from the same root.

2.4 TRAINING AND TESTING RESOURCES FOR SPEECH RECOGNITION

In order to develop high performance speech recognition systems, an extensive set of pre-computed audio and text resources are needed. In order to train the HMM, DNN or GMM a high amount of transcript speech is needed. In order to compute n-gram probabilities, a text corpus as large as possible is needed. Although there are a wide variety of training and testing resources for speech recognition, we will mention some of the most important used as references for validating new algorithms.

The [TIMIT, 1993] contains a large corpus of read speech dedicated for acoustic-phonetic studies and development of LVCSR. The database was created with the help of 630 speakers and is designed for 8 major dialects of American English. The database was created as a joint effort between the Massachusetts Institute of Technology, SRI International and Texas Instruments. The database was used for computing the recognition performance of the systems mentioned in table 2.1.

Another well known database is [AN4, 1991]. It was recorded at Carnegie Mellon University in 1991. It is an alphanumeric database and was used in several theses over the years.

The Linguistic Data Consortium (LDC) distributes a fairly large number of databases. The first 10 databases considering the frequency of downloading are mentioned. TIMIT – mentioned above is the most downloaded database. The Web 1T 5-gram contains English word n-grams and their observed frequency counts. The CELEX2 holds lexical databases of English, Dutch and German and was developed as a joint enterprise between German and Dutch research centers. TIDIGITS was originally developed and collected at Texas Instruments and contains sequences of connected digits. It is developed with the help of 326 speakers (children and adults). Treebank-3 is another important database that holds 2500 stories from Wall Street Journal along with the raw texts. The New York Times Annotated Corpus contains more than 1.8 million articles that were written and published by the New York Times between 1987 and 2007. The database is useful mainly for language models building. Other databases for language model development are TIPSTER and American National Corpus. NTIMIT is also an interesting database to download as it contains all the original TIMIT recordings transmitted through a telephone handset with various channels in the NYNEX telephone network and re-digitizing them.

The largest speech and text corpus for Romanian language was described in [Cucu, 2011] and was created within the Speech and Dialogue Research Laboratory at University "Politehnica" of Bucharest.

2.5 SPEECH SOURCE SEPARATION

One of the main challenges of a LV-CSR is caused by environments with competing speakers. In this case, it is difficult for the system to firstly select the speaker of interest and then perform a speech recognition task. The selection of the target speaker is called speech source separation. This is a topic widely studied for automatic speech recognition. There are several methods that deal with this challenge but they will be discussed in detail in chapter 4. Most of them use microphone arrays but there are some that also use a single microphone.

2.6 ACCENT, DIALECT AND EMOTION INFLUENCE

When a speaker that is using a LV-CSR system has an accent he might experience poor recognition performances if the system is not designed to detect the accents. Accents generate 2 issues with LV-CSR systems. The first challenge is that they need to be detected. If the task of detecting a succession of phones is difficult, attempting to decide if a specific sentence is pronounced with an accent is a challenge even more difficult. The second problem is that when the accent is detected, the system needs to know how to treat the accent and consequently it needs an additional acoustic model to use. Therefore, accents greatly increase the complexity of LV-CSR systems. There are research studies that treat the accent problem in speech recognition.

[Fung, 1999] describes methods for fast accent identification and the measures taken when performing speech recognition when accents are present. The authors use an accent adapted dictionary that results in a reduction of PER with 13.5%. [Zheng, 2005] presents methods for accent detection and speech recognition for Shanghai accented Mandarin language. The study uses a database of spontaneous speech collected from 50 male and 50 female speakers with accents. While the traditional methods for accent detection used HMM-GMM classifiers of MFCC data to decide if a speaker has an accent, [Zheng, 2005] studies the effects of accent on the phones pronunciation and create a metric that groups speakers into "more standard" and "more accented" categories. The metric is based on the fact that accent leads to the replacement of standard retroflex fricatives and affricates with their alveolar equivalents.

[Biadsy, 2011] is an extensive study over the methods used in dialect and detection. It describes various approaches for dialect detection like: phonotactic modeling, prosodic modeling, traditional acoustic modeling, discriminative phonotactics and kernel based methods. Phonotactic modeling exploits the fact that dialects differ in phone sequences distributions. The approach presented in [Biadsy, 2011] runs multiple phone recognizers in parallel – trained for different accents and dialects – and outputs the result of the recognizer with highest likelyhood. The prosodic modeling exploits the fact that dialects differ in prosodic structure. Therefore, the author defines a set of features that describe the prosodic structure of a recognized speech, and using a classifier if decides what is the input dialect and accent. Acoustic modeling relies on the fact that dialects differ in spectral distribution. The last approaches are a combination of the previous three exploiting simultaneously differences in phonetic, prosodic or acoustic structure.

The speaker's current emotional state plays an important role in speech recognition. This complicates even further the design of a LV-CSR because the system has to be aware of the user's emotional state with two possible follow-ups: one is to notify the user that it is likely to generate erroneous results or the other one is to use an acoustic model "aware" of changes in emotional state. Like in the case of accents and dialects, emotion recognition is also treated in some research studies. [Vogt, 2005] is a framework that tries to determine emotions using acoustic properties of speech. It uses an underlying database of reference emotions used for training. [Ayadi, 2011] presents a survey on the speech emotion recognition methods, touching aspects related to feature extraction, classification methods and existing databases. [Han, 2013] states in a presentation made for Microsoft Research that despite the great progress made in artificial intelligence, a natural interaction between a person and a machine is far from being accomplished because the machine cannot recognize the person's emotional state and react in consequence.

2.7 JARGON FLEXIBILITY

The usage of jargon is a challenge for creating an efficient language model. While detection of jargon does not have direct applications in the field of speech recognition but rather related to surveillance jargon usage still needs to be considered when creating a language model for a speech recognition engine. [Pal, 2013] presents a set of methods for detecting jargon words in e-Data using semi-supervised learning. The algorithms use various approaches by trying to determine the sense of a word and identify if a suspicious word belongs to the context. The approach goes even further to replace the jargon words with suitable candidates.

The effect that jargon would have when used in a speech recognition software is that the probability of using a sequence of words containing a "slang" word would be very low considering n-gram probabilities computed on text corpora without jargon. Therefore, the language model needs to be aware of jargon existence to fine tune its n-gram probabilities.

2.8 SUMMARY

Chapter 2 was an overview above the most widely used techniques for automatic speech recognition. The focus of the chapter was to highlight the two main feature extraction methods, MFCC and PLP and to show how they are used for phone recognition. Regarding phone recognition we presented HMM-GMM methods along with newly developed DNN methods. We observed that many state of the art speech recognition systems produced by Microsoft and Google use DNN based methods. We also observed that the phone error rate for many reference LV-CSR systems is around 20%, error that propagates when it comes to word or sentence error rate. Therefore, we highlighted an alternative path to classic statistic methods for phone recognition, which is using spiking neural networks. While this alternative is far from yielding results of traditional methods, it may be a viable future method as the third generation neural networks are able to simulate much better the interactions between neurons in the human brain because they also model the synapses. We also presented briefly some techniques for language modeling mainly using n-grams and morpheme based language models. We observed that morpheme based language models are more suitable for languages with sparse resources and "out of vocabulary" words. We also highlighted some of the additional challenges that increase the complexity of a LV-CSR which are accent, dialect, emotions and jargon. We stated that accent and dialect based speech recognition need to use multiple acoustic models for each targeted dialect or accent. Detection of accent or dialect is also a challenging task and existing method generally exploit the specifics of each accent in phonetic, prosodic or acoustic structures. Emotion detection also exploits the acoustic structure. We also stated that jargon is a challenge for creating a language model because the usage of jargon can make the system generate a wrong word because the probability of using "slang" words is low when computing n-grams.

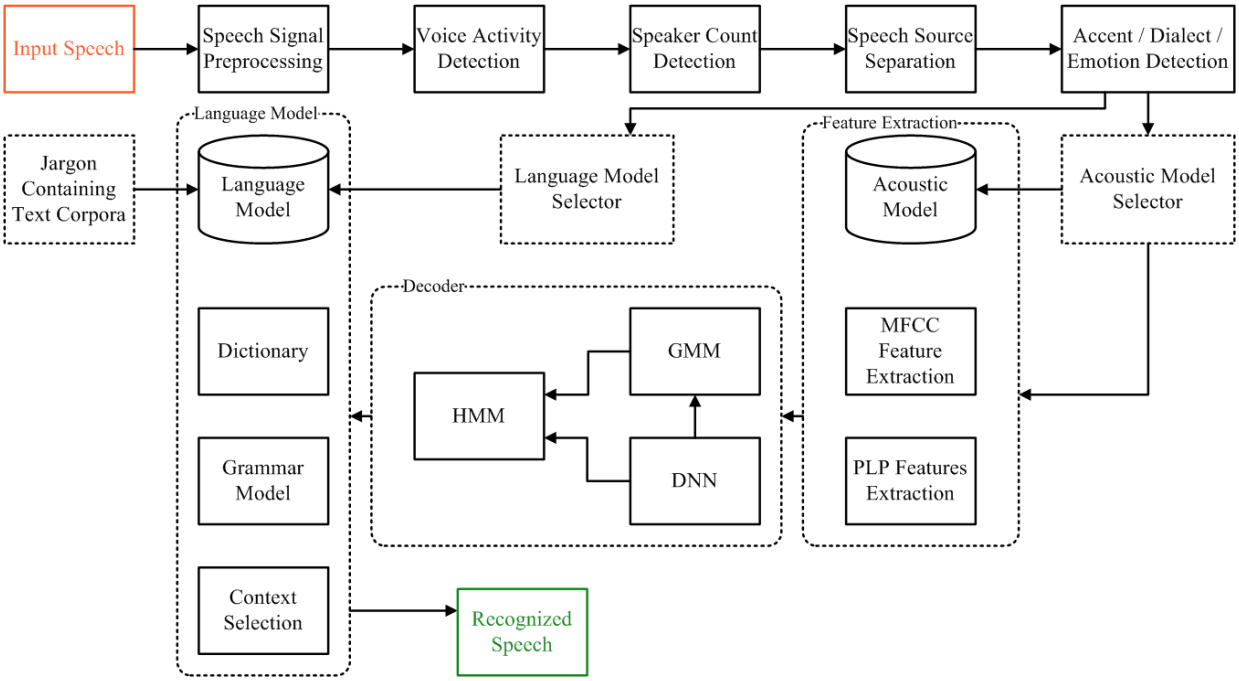

Figure 2.6 – A diagram proposal for a complex LV-CSR

In figure 2.6 some new components have appeared when comparing it with figure 2.1. The diagram is more complex and integrates the mentioned methods and algorithms and it also contains additional elements that will be treated in the following chapters. The voice activity detection and the speaker count detection will be treated in detail in the following chapter. One of the main contributions of this thesis is the development of a method that is able to determine the speaker count in a competing speaker environment. This is particularly useful for methods of blind source separation, described in chapter 4 which need the number of competing speakers as an input parameter, before attempting to separate speech sources.

3 VOICE ACTIVITY DETECTION

Voice activity detection (VAD) is generally no longer a technical challenge for speech assisted systems. This is true for cases where a single speaker is talking. For cases with competing speakers voice activity detection is necessary to be accurate but in an ideal case this processing step would also output the number of competing speakers. In the following paragraph we will treat both the single speaker and competing speakers cases and we will describe an original method for determining the speaker count in a multi-speaker environment. When referring to the single speaker cases we will highlight some of the challenges that occur in this space.

3.1 SINGLE SPEAKER VOICE ACTIVITY DETECTION

A very simple method for voice activity detection is based on determining the signal's power within a timeframe. We can observe in figure 3.1 that there is strong correlation between the periods of speech and the signal's power. This is a basic method and can yield reasonable results in environments with no perturbation. Another advantage is that it has *O(N)* computational complexity and it is very cheap to be implemented either in hardware or software.

The method mentioned above has some disadvantages that are the price paid for the simplicity of the approach. The biggest challenge is noise. The method cannot perform accurate in noisy conditions. If the noise is not white it can be removed by computing the signal's power in a selected band – typically from 300 Hz to 4 kHz which is associated to human speech frequency band. If the noise is white there are other approaches to eliminate it.

Another technique used to improve the accuracy of voice activity detection is window overlapping. This is mainly done to remove the "edge-noise" due to segmentation in the time domain. In [Prasad, 2002] the authors categorize the VAD methods into frequency based and time based analysis approaches. However, the focus of the study is to optimize data transfers in telephony and, therefore, the simplicity of the algorithm is essential.

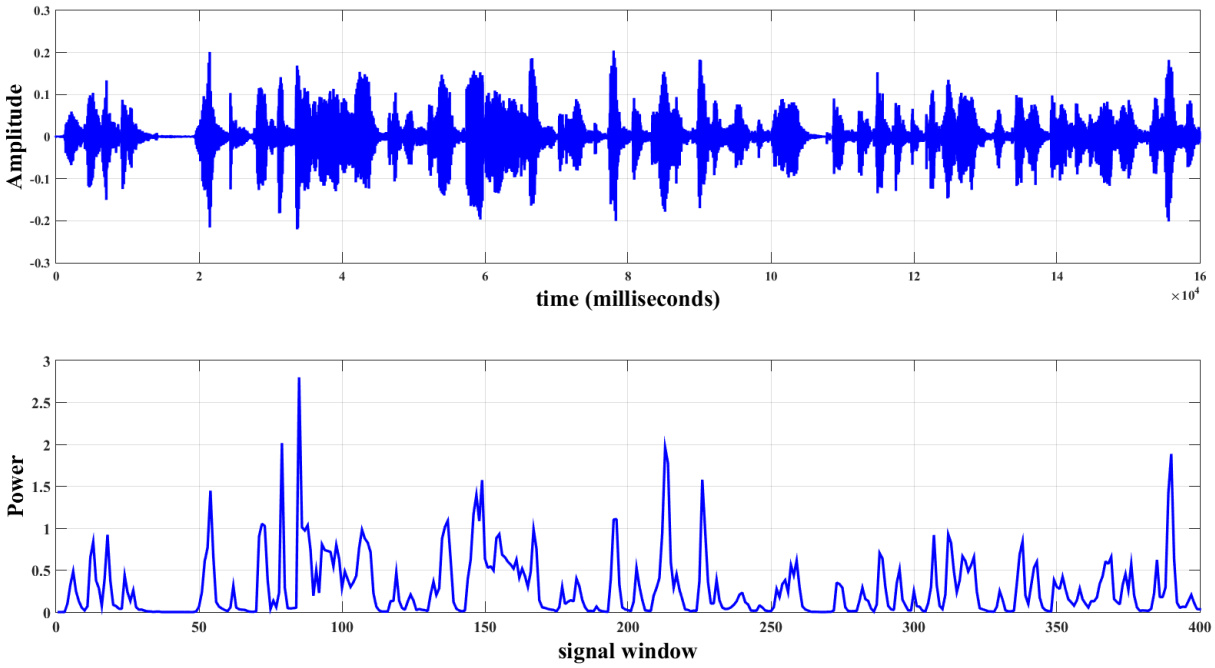

Figure 3.1 – Correlation between periods of speech and signal power

3.1.1 Adaptive supervised learning based VAD

For speech recognition assisted systems, the simple time or frequency analysis approaches might not provide sufficient accuracy. This is why a viable alternative for the VAD challenge is using a training approach based on a set of *silence and noise models*. There are multiple variations of VAD methods based on adaptive supervised learning but they tend to respect the flow described by figure 3.2.

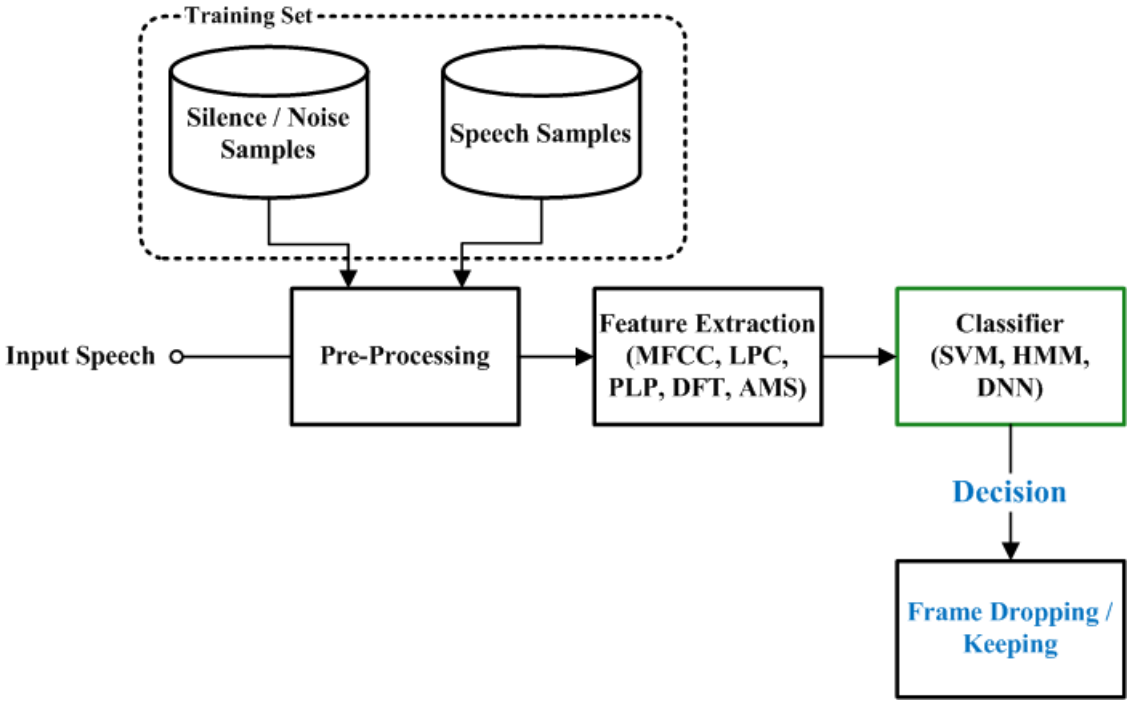

Figure 3.2 – VAD based on adaptive learning methods

The general idea is to train a classifier to decide whether a speech frame contains speech or is just noise or silence. Therefore, a training set of silence, noise and speech samples is required. Ideally the training database should contain multiple noise models. The classifier would yield superior performance if it would operate on feature vectors rather than plain input signals. In [Shin, 2010] the authors define a set of feature vectors by estimating 3 types of SNR: a priori, a posteriori and predicted. Then a generalized gama distribution is being defined for likelihood estimation in order to adapt the classifier coefficients. [Zhang, 2013] introduces feature fusion which involves computing multiple sets of feature vectors like: DFT, MFCC and LPC coefficients. Additionally the method uses relative-spectral perceptual linear predictive analysis coefficients (RASTA-PLP) introduced by [Ellis, 2005] and coefficients generated by amplitude modulation spectrogram (AMS) discussed in [Tchorz, 2003] and [Kim, 2009]. [Kinnunen, 2007] proposes a VAD with good accuracy by using only MFCC coefficients.

The feature classifier can vary from one implementation to another. For example, [Dong, 2002], [Kinnunen, 2007] and [Shin, 2010] use all support vector machines (SVM) implementations while [Zhang, 2013] proposes an accurate classifier based on deep belief neural networks. The VAD systems can be also implemented using HMM objects by the exact same process that is used to train phone decoding.

3.1.2 Non adaptive learning based VAD

The methods described in the previous section need the existence of a large set of labeled data. This can be a laborious process and therefore unsupervised methods for classifying signal frames would be preferred. In order to demonstrate progress compared to time or frequency based basic analysis used in telecommunication systems, the proposed VAD's should respond well in heavy noise conditions.

One such system is described in [Germain, 2013]. The authors proposed a method inspired from blind speech source separation research field for classifying input signal frames. The feature extraction step generates a set of short time Fourier transform (STFT) coefficients that are fed into a processing stage where a non-negative matrix factorization is being computed. The resulting *activation coefficients* are summed and passed through a filtering stage and finally compared to a threshold determined experimentally. The method is stated to outperform some of the referenced VAD methods.

Another method that does not use supervised learning is proposed in [Ramirez, 2004]. A long-term spectral divergence (LTSD) metric is introduced, which describes the divergence between speech and noise, and it can lead to a classification rule with high discrimination. An important component of the LTSD metric is the defined long-term spectral envelope (LTSE). The LTSE is derived from the set of overlapped frames of the speech signal, as described by (3.1). Then the LTSD metric is computed using (3.2).

$$LTSE_N(k,l) = \max\{X(k,l+j)\}_{j=-N}^{j=+N} \tag{3.1}$$

$$LTSD_N(l) = 10\log_{10}\left(\frac{1}{NFFT}\sum_{k=0}^{NFFT-1}\frac{LTSE^2(k,l)}{N^2(k)}\right) \tag{3.2}$$

After the LTSD is computed a threshold based decision rule is applied to determine if the current frame will be marked as noise or speech. The LTSD is also a measure of confidence. Therefore, if the LTSD is too low, an additional step called *hang-over scheme* is applied, which uses the previous observations to produce the final decision.

3.2 VAD IN COMPETING SPEECH ENVIRONMENTS

All the methods mentioned in section 3.1 operate on input signals where a single speaker is active. The approaches just tag the frame to be noise or speech. In a competing speech environment which is more close to real environments, valuable information would be the number of speakers that talk simultaneously. This can enable superior accuracy to next stages in a LV-CSR system as proposed in figure 2.6, like the speech source separation stage.

3.2.1 VAD for multi-speaker setups using cross-talk analysis

In this paragraph we will discuss about some of the existing methods that address the VAD task in multi-speaker environments. One of the first challenges in such an environment is represented by crosstalk. For example, in [Pfau, 2001] by using a VAD system presented in figure 3.3, the VAD results are stated to be inaccurate unless a post processing stage where speaker overlap is analyzed is not present. The method uses a microphone for each of the speakers but when speakers are talking simultaneously, they also get recorded on microphones assigned for other speakers – of course with lower intensity. The main issue is that due to crosstalk, the VAD might trigger a speech period for a speaker who is not actually talking.

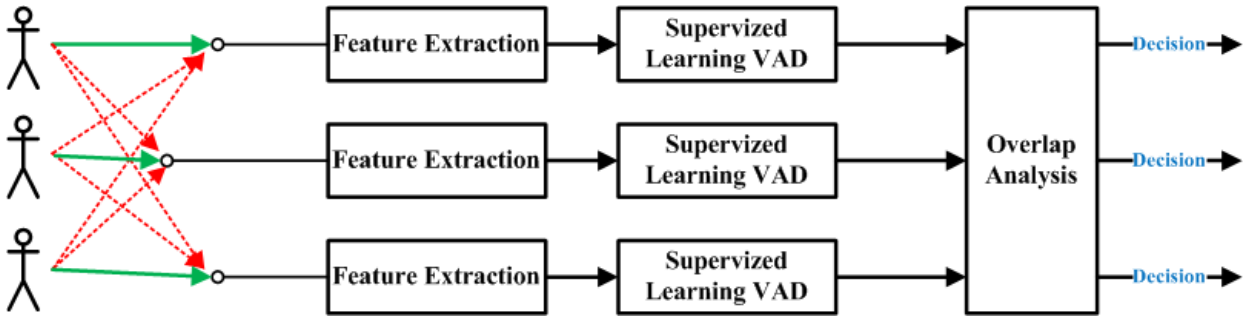

Figure 3.3 – Crosstalk influence in multi-speaker environments

[Pfau, 2001] uses a 25 dimension feature vector composed out of the loudness computed for 20 critical bands on the Bark scale up to 8 kHZ, the total energy of the signal frame, the total loudness and a modified loudness metric along with the zero crossing rate and the difference between the energy captured by the closer microphone and the averaged energy captured by the distant microphones. This feature vector is classified by an HMM-GMM based model. The overlap analysis is the essential step in improving the accuracy of the VAD in crosstalk environments. In order to achieve that, the authors introduce the peak normalized short-time cross-correlation between the active channels *i* and *j* as described by (3.3).

$$\hat{\rho}_{ij} = \max\left\{ \frac{\sum_n x_i[n] \bullet x_j[n+1]}{\sqrt{\sum_n x_i[n]^2 \bullet \sum_n x_j[n]^2}} \right\} \tag{3.3}$$

The short-time cross-correlation is expected to be lower for cases where two or more speakers are talking simultaneously, while for cases where a single speaker is talking but due to crosstalk it fires incorrect VAD decisions for distant microphones, the metric is expected to be higher. Based on this, a final decision can be issued to remove some of the false VAD

triggers. The proposed method by [Pfau, 2001] is stated to produce a WER lower than 10% when integrated with a speech recognition system.

3.2.2 VAD for multi-speaker setups using source localization

Another algorithm that addresses VAD in multi-speaker setups is proposed in [Maraboina, 2006]. The setup is very similar to the one described by figure 3.3. The original part of the work is the overlap analysis block. An important difference is that the authors use a simple energy-based processing as preliminary VAD. This removes the need for supervised learning making the method more robust to environments that were not touched by the training set. After the preliminary VAD, the frequency spectrum is being computed for all the inputs incoming from the sensors. The next step is applying an independent component analysis (ICA) in frequency domain. Since ICA approaches are more related to blind speech separation, they will be presented in chapter 4 of the thesis. In order to be able to apply ICA the authors simulated a microphone-array setup in a reverberant room where speakers were located at 1m distance from the microphones. The ICA outputs a *de-correlation matrix* that can be used to separate the sources. This makes the method useful for cases where BSS is also targeted. The beam-pattern analysis is the next processing step for determining the overlap periods of the speakers. This type of analysis estimates the direction of arrival for speech sources. When used by [Maraboina 2006], the algorithm determines the number of frequency components that arrive from the estimated directions of each source. Finally, the output of the beam-pattern analysis is being used for labeling the active parts of each speaker.

Another technique that addresses VAD in multi-speaker environments is proposed by [Chen, 2012]. The approach uses spatial localization of speakers, beamforming and only after the VAD comes into place. Only 2 speakers are used by the authors so this is one of the drawbacks of the method. We expect that the computational complexity would scale worse than linear when increasing the competing speaker count.

The approach uses a set of microphone arrays to firstly determine the spatial location of each of the 2 speakers. The most common method for determining the location of a sound source is determining the time difference of arrival (TDOA) which can be studied from [Gustafsson, 2003]. When using 2 sensors, the source direction can be obtained. When having more than 2 sensors triangulation is possible. In order to determine the τ argument the cross-correlation metric between the signals captured by 2 microphones needs to be determined using (3.4).

$$R_{x1,x2}(\tau) = \sum_{n=-\infty}^{\infty} x_1(n)x_2(n+\tau) \tag{3.4}$$

In the above formula, the highest level of correlation implies that the τ is closest to the TDOA metric. One widely used example of TDOA is called interaural time difference and is expressed by (3.5), where τ is the time difference expressed in seconds, d is the distance between the microphones in meters while v_s is the speed of sound in m/s and θ represents the angle between the median of the microphones and the sound direction.

$$\tau = \frac{d \sin\theta}{v_s} \tag{3.5}$$

Using (3.5) and at least 3 microphones, the sound source can be determined as illustrated by figure 3.4. It is observable that as the number of sources grows, the localization is likely to be more accurate. This approach is used by [Chen, 2012] as a first step in implementing the multi speaker VAD. The next processing element is beamforming.

Beamforming is essentially an approach to set the directionality or the direction of reception for a transmission. This is a widely used technique in telecommunications. Also, some high-end premium smartphones have started using beamforming techniques for improving the quality of speech assisted tasks.

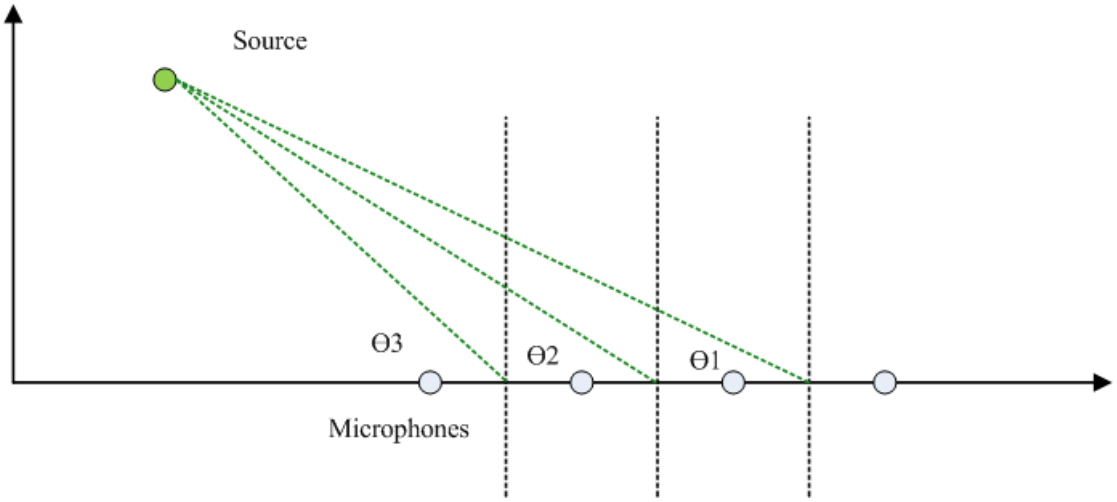

Figure 3.4 – Sound localization using TDOA

The configuration proposed by [Chen, 2012] in order to minimize the mutual information between channels is called generalized sidelobe canceller (GSC). The goal is to maximize the amplitude of the signal from a selected direction, corresponding to one speech source. The beamforming stage is stated to bring value to the source separation methodology but is admitted being far from perfect therefore there is the risk that the system will issue false statements due to the residual speech present in the target beam. In order to minimize this risk, the authors also measure the cross-meeting normalized energy to analyze the amount of overlap between speakers.

The method proposed by [Chen, 2012] was analyzed from the performance point of view after being integrated with a word recognition system, and compared to a manual segmentation process. The word error rate (WER) was 39% for manual segmentation and 41.4% for the discussed approach. We suspect that this is the cause of extensive overlap between speakers that cannot be fully separated and therefore inject high amounts of errors into the phone decoding stages of the word recognition system.

4 ESTIMATING COMPETING SPEAKERS' COUNT

Auditory scene analysis (ASAN) is a human reflex ability allowing us to distinguish multiple speech sources in a competing speaker environment, and focus our attention towards the speaker of interest (selective auditory attention – SAA). The challenge of developing computer based methods for ASAN has been accepted by a fair number of researchers and can yield useful applications in contexts not limited to ambient assisted living, forensics, blind source separation, smart surveillance and automated data mining. Taking decisions using ASAN related methods is a difficult task due to the complexity of the mixed speech signals. In addition, there is no sufficient understanding on how human SAA works; in order to develop biology inspired methods.

One essential information extracted using computer based ASAN is the number of speakers producing speech at a certain moment in time. This process can follow or integrate with the voice activity detection processing stage and add a new functionality: *the ability to output the active speakers count per analysis frame*. All the methods discussed above for voice activity detection do not output the number of concurrent speakers and generally operate on low speaker counts. We consider that having an algorithm for estimating the number of speakers talking simultaneously is of great value for improving the accuracy of speech recognition in multi speaker environments. One such benefit is that knowing the competing speakers enables a class of methods that perform blind speech source separation. These methods need the number of competing speakers in order to perform the separation. In addition determining speaker number in multi-speaker environments has potential applications in the fields of surveillance, audio classification, ambient assisted living and others.

We will describe in the following paragraph a collection of experiments done in order to develop a set of methods that would be able to estimate the speaker count in a single microphone recording. This method could have an increased adoption potential as it does not require multiple sensors, making it applicable for a wide range of devices from smart-phones to low power embedded devices. The same algorithms can find applications also in blind speech separation systems that use more than one sensor for recordings.

4.1 STATE OF THE ART SYNTHESIS

The value of being able to output the number of competing speakers in the early stages of speech recognition has been seen by several researchers. In [Rivet, 2007], [Almajai, 2008] and [Minotto, 2014] the authors propose a visual voice activity detection. These studies utilize numerous concepts from the image processing research field in order to determine when a speaker is talking by following the lips movement. Such systems are typically a variation of the architecture proposed in figure 4.1.

There are implementation related specifics at the level of each computational block. [Minotto, 2014] proposes the usage of optical flow methods implemented using Lukas-Kanade algorithm for extracting the features of the tracked frame in the visual stream. The video classification is based on support vector machines (SVM). The method uses also HMM models for silence and speech, which are taking as input acoustic features like MFCC.

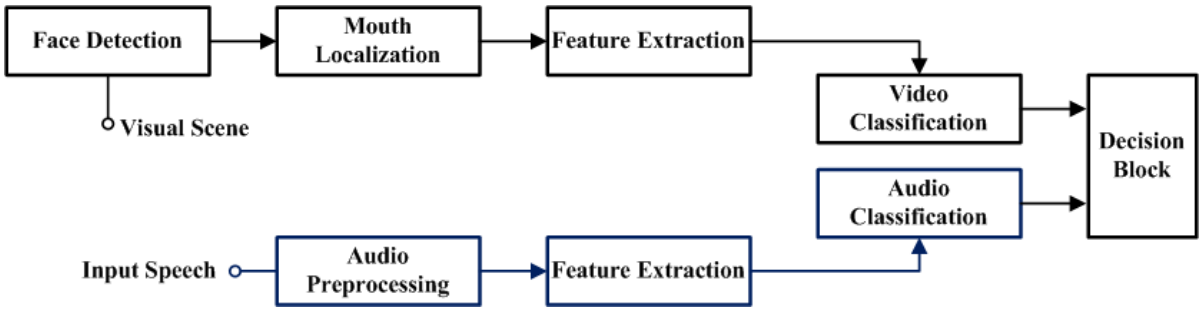

Figure 4.1 – Visual-voice activity detection methods processing chain

The implementation in [Almajai, 2008] is simpler in the sense it uses a GMM classifier for both visual and audio features. The audio features are classical MFCC while for the visual vectors a 2-dimensional discrete cosine transform (DCT) is being computed.

[Rivet, 2007] does not focus on describing the audio feature extraction and VAD but describes in detail the method used for visual voice activity detection, V-VAD. The paper considers the open mouth as an ellipsis with varying width and height. Therefore, *w(t)* and *h(t)* can be used to derive a set of dynamic coefficients. If *w(t)* and *h(t)* are within the thresholds established in order to decide that the tracked person is actually speaking, this information can pair the response from the audio VAD and produce superior VAD performance. However, the declared goal of [Rivet, 2007] is to contribute to the blind separation of convolutive mixtures and therefore with increased computational complexity, the method can be used to roughly estimate the phone class being pronounced at a certain moment by using the dynamic coefficients. By phone class we refer to vowels or consonants categories which might improve the speech source separation attempt.

There are other attempts that use only the audio signal properties for estimating the number of competing speakers. For example, [Sayoud, 2010] introduces a metric that is stated to be an estimator of the concurrent speakers count. The metric is derived from the 7^{th} coefficient of the MFCC set and defined in (4.1).

$$PENS = \overline{mel_7} - \sqrt{\mathrm{var}(mel_7)} \tag{4.1}$$

This is an extremely simple estimator but the method yields more than 40% estimation error for speaker counts above 4. [Arai, 2003] also defines a new term that is stated to be correlated with the concurrent speakers' number, based on the modulation characteristics of

speech. The authors have observed a distinct modulation peak pattern that is expected to decrease as the number of speakers increases. The work is one of the first that address this challenge but due to its simplicity it cannot scale well with increased speaker count and this reduces the number of applications in which it can be adopted.

4.2 INSTANTANEOUS SPEECH MIXTURES DATABASE

In order to test the potential of each feature extraction method, we need a set of speech mixes. Figure 4.2 describes a method we used for creating the mixtures database. We used a set of 15 recordings collected from 15 male speakers with ages between 22 and 40. We used only male speakers to avoid potential misleading results when combining male and female voices. All the recordings were created in a soundproof room that minimizes reverberation using a high quality recording device.

Using the 15 recordings there is a high number of combinations we can use to create the mixtures. For most of our experiments regarding the feature extraction parameter studies we created 10 mixtures for each speaker count. Each of the mixtures have different speakers combination except where we have high numbers of speakers participating, thus not allowing for 10 distinct combinations – mixtures with more than 13 speakers. The single speaker recordings were also used for perception studies discussed in future sections of this thesis and, therefore, we will describe more the content of the recordings where it shows more relevance to the topic.

Each recording had 150 seconds. We generated the speech mixtures using less seconds. Most of our experiments used 10 seconds of speech. Therefore, we had the possibility of extracting randomly a sub-signal of 10 seconds from the original recording adding more coverage for the data spectrum.

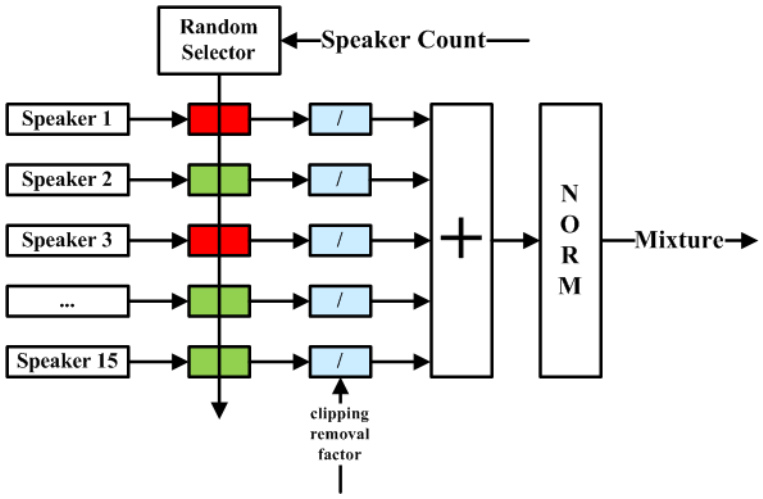

Figure 4.2 – Speaker selection when creating mixtures

$$\mu_{x_m} = \frac{1}{N}\sum_{i=1}^{N} x_m$$
$$S = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left|x_m - \mu_{x_m}\right|^2} \tag{4.2}$$
$$x_{t1} = x_m - \mu_{x_m}$$
$$x_{t2} = \frac{x_{t1}}{S}$$

The signals determined by the random selector in figure 4.2 were mixed using the formulae defined in (4.2). We multiplied each source with a clipping removal factor, to remove the undesired clipping effects that would have occurred when adding the sources. Clipping would have occurred in case of overflowing numerical limits. We can observe that an additional normalization step was performed. The normalization is however done on the mix and not on the sources. If it had been done on the sources that would have over-trained the method for particular cases where speakers are equally distanced to the microphones and are speaking with equal loudness. Our intention is to develop a method that would cover as many real use-case scenarios as possible. This mixing model will be used throughout all the experiments that target simultaneous speaker counting.

Unfortunately, this mixing model fails to map reverberation related behaviors in the presence of many speakers that occur due to sound reflections in a room. Creating a set of mixes up to until 15 simultaneous speakers is a laborious task if a real environment is used but we expect that ignoring some of the mentioned effects will not impact the meaningfulness of the proposed methods. In addition, a similar mixing approach was also used by [Arai, 2003] and [Sayoud, 2010] in their proposed experiments.

The following subsections will present how close is a certain feature extraction methodology to respecting the objectives described in the beginning of this section.

4.3 FEATURE EXTRACTION APPROACHES

In this section we try to discover a set of parameters that can be used to estimate the number of competing speakers. We are interested in a set of features that would increase linearly – or just evolve within a predictable trend – with the growth of competing speakers. Ideally the trend would be linearly allowing for a set of thresholds marking each count. It is obvious however that in order to generate the set of thresholds a high number of recordings need to be used in order for the determined values to be statistically relevant. We also need to take into consideration different noise configurations when determining the set of thresholds.

Another fortunate case would mean that the feature is easily to be computed resulting in increased adoption possibility. We mainly desire that the feature extraction approach should be able to function in real-time ensuring a good user experience indicator.

Nevertheless, the feature extraction method should provide sufficient separation for reduced signal sizes, allowing for accurate estimation of speaker counts thus improving the performance of the following blocks in the LVCSR system.

So, to summarize, we desire that the feature extraction step would generate a set of parameters that need to respect as much as possible the following rules:

- Evolve within an observable trend correlated with the evolution of the number of competing speakers;
- Allow a fast implementation considering the computational complexity;
- Provide sufficient separation degree even for reduced signal frames.

4.3.1 One dimension features

The simplest method for determining the number of active speakers in a timeframe is having a single value feature and comparing it with a threshold. This yields the lowest computational complexity. However, a single value metric cannot contain information about many of the characteristics that occur in speech mixtures. Therefore, we expect that the classification potential for such a metric to be low. In the following sub-sections we will discuss some potential single value features that can provide information about the number of competing speakers.

4.3.1.1 Estimation based on signal power

We would expect that the power of a signal would increase directly proportional with the number of speakers creating the speech signal. This metric also has the advantage of simplicity considering the above rules. We want to investigate if it can demonstrate good separation for increased speaker counts and good separation accuracy for size reduced frames. We observed in section 3.1 that this parameter can provide reasonable performance when used in voice activity detectors but we are evaluating if it can also contain information about the number of speakers talking simultaneously.

In our first evaluation we tried to understand if the overall power carried by a signal is expected to grow when the number of speakers increases. We also aimed for determining if this supposed behavior is influenced by the observation window size. We used 10 different window sizes ranging from 0.5 seconds to 5 seconds, with a step of 0.5 seconds. Considering that for each speaker count we have a set of 10 mixtures, we randomly selected 2000 different windows from the 10 mixtures, with the corresponding number of samples, for which we computed the normalized power using (4.3).

$$P(x_m(k)) = \frac{\sum_{i=1}^{N} x_m(k)^2}{2N+1} \tag{4.3}$$

The first experiment produced a set of normalized powers considering varying frame sizes for speakers count ranging from one to 15. Figure 4.3 shows on the X axis the window size in seconds while on the Y axis it displays the normalized power computed for that particular window size. Each line in the chart represents a specific speaker count. As expected the bottom line highlights the power levels computed for single speaker recordings.

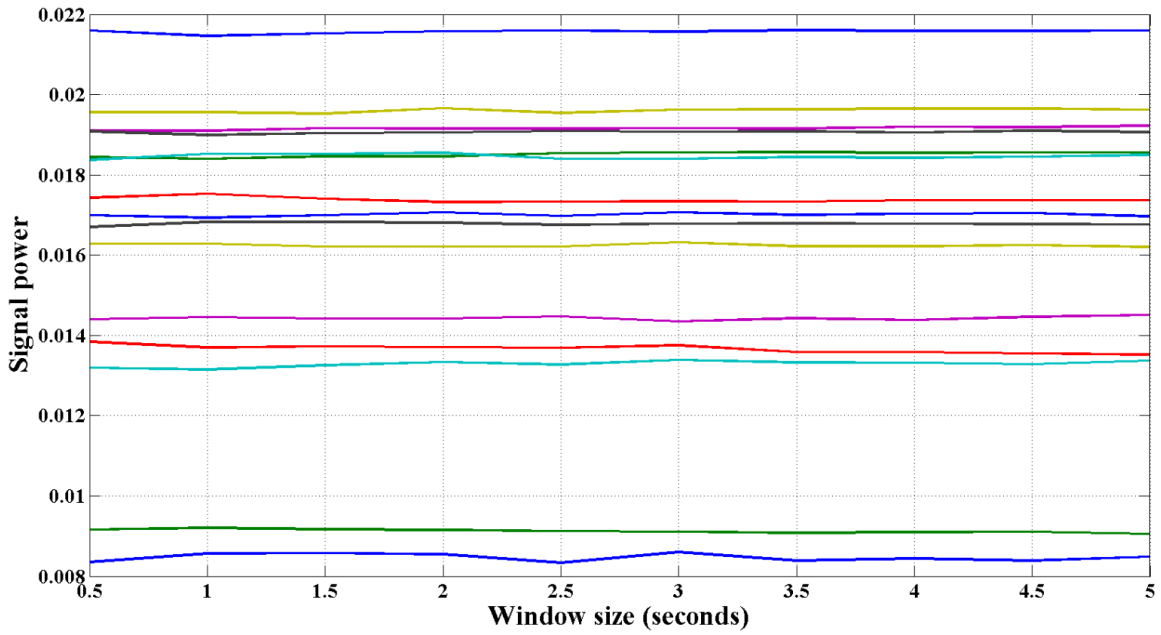

Figure 4.3 – Power level used as speaker count estimator

Figure 4.3 tells us that the power level carries some information about the underlying number of competing speakers. We see that the value of the normalized power level generally increases with the number of competing speakers. We see clear separation for speaker counts of one and two. However, for higher speaker counts the separation is not that obvious. In addition we see an overlap when considering 12 and 13, or 10 and 11 simultaneous speakers. This indicates that the method might not scale with an increased speaker count.

Figure 4.4 highlights the trend of the power level with the increasing number of speakers. We see that this trend is not necessarily monotonic and this fact highlights a weak separation performance for this method. Due to the fact that the trend cannot be modeled analytically with an elementary monotonic function it is difficult to establish a set of thresholds associated to each speaker count.

In addition the power levels are an average of multiple experiments. In figure 4.5 we have selected a 0.5 seconds window and selected a set of random windows from each of the 10 mixtures associated to each of the 15 speaker counts. As in figure 4.3 the horizontal series represent a certain speaker count. We are interested to observe if the signal powers for two or more competing speakers show a degree of overlap. As figure 4.5 shows we see a great amount of overlap meaning that detecting the number of simultaneous speakers is impossible for a 0.5 seconds window. Figure 4.6 – where we selected an analysis frame of 45 seconds – shows that the overlap between power levels decreases drastically as the size of the analysis window increases. However, we should not expect to get lower overlap than the one reflected in figure 4.3, which contains averaged values of the signal power.

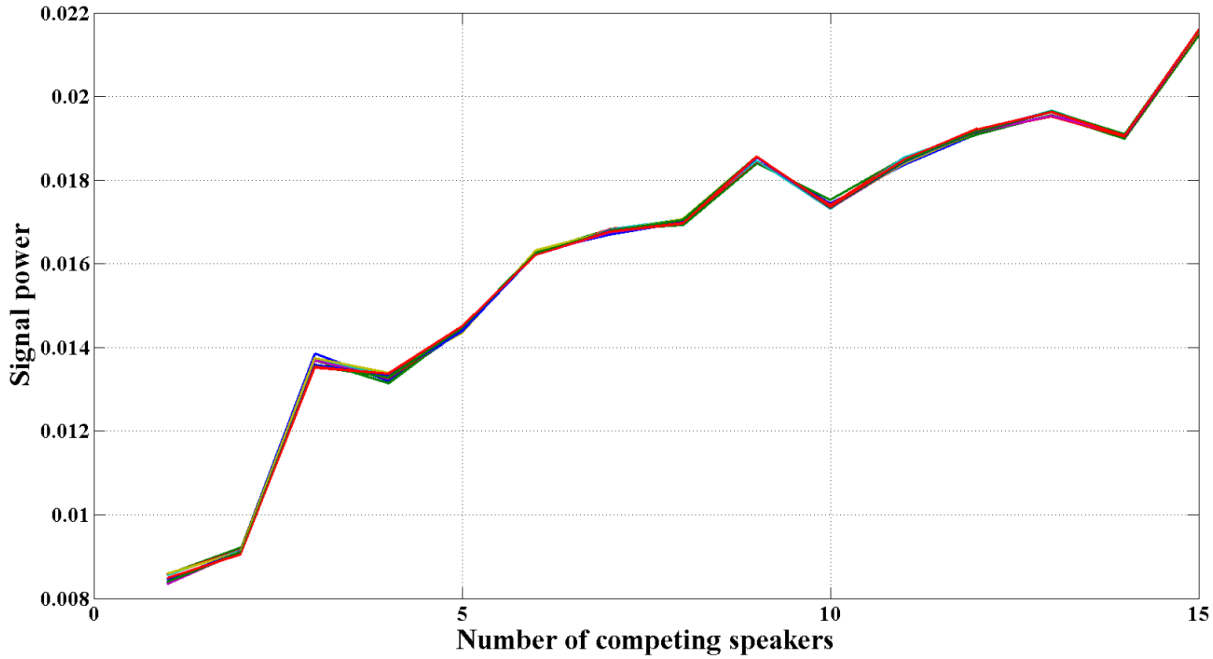

Figure 4.4 – The power level trend with the increasing number of speakers

Figure 4.6 displays additional information besides the decreasing overlap between power levels when using a longer analysis periods. It shows that the power levels vary greatly from one mixture to another. Each horizontal series as an associated set of 200 signal windows where from each of the 10 mixtures a number of 20 windows were selected. We can see that the signal power differs from one mixture to another, even for identical competing speaker counts. In other words the method is sensible to the voice specifics that participate in each mixture. This is a drawback that needs to be solved for the method to show adoption potential.

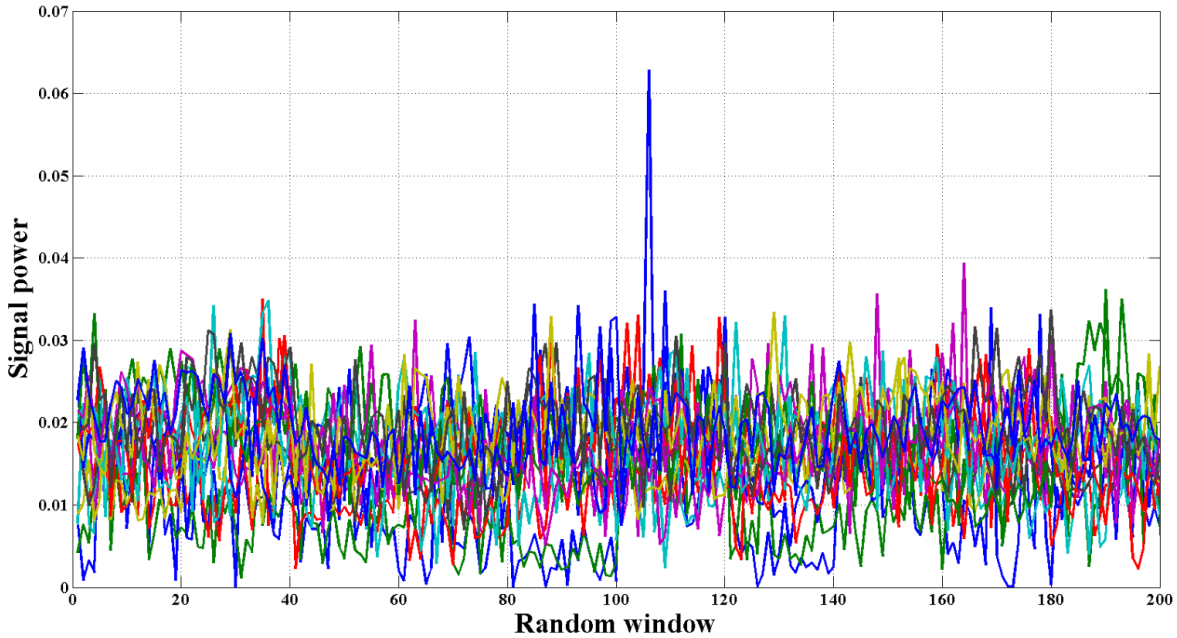

Figure 4.5 – Overlapping power levels for different competing speakers count

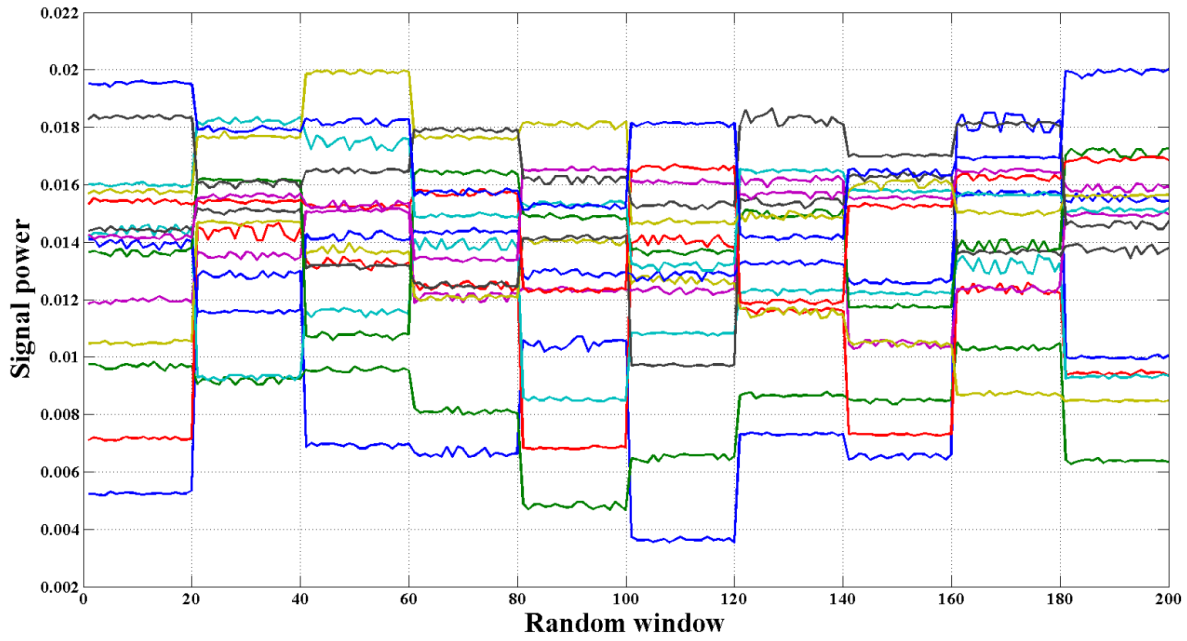

Figure 4.6 – Variation of power level depending on speech mixture

Judging by the overlaps present in figures 4.5 and 4.6 we can only say that the signal power holds some information about the number of competing speakers but this information is not sufficient for accurately determining the count in a short timeframe. All we can say is that on average, the signal power is expected to grow, as the number of competing speakers increases.

4.3.1.2 Estimation based on signal zero crossings

Another widely used metric for VAD is the number of zero crossings for a timeframe. In the context of speech mixtures we want to observe if this holds some valuable information to help us distinguish number of active speakers. The zero crossings of a signal can be computed straight forward with O(N) complexity, thus not raising speed related issues. In our experiments we used the zero-crossings rate computed using (4.4), where L is the number of samples of the timeframe.

$$ZCR_{x_m} = \frac{\sum_{i=2}^{L}(x_m(i-1)\bullet x_m(i) < 0)}{L} \tag{4.4}$$

Let us study the classification potential of ZCR. In figure 4.7 we used two different window sizes of 0.5 seconds and 45 seconds. We can clearly observe that for 0.5 seconds timeframe, there is practically no separation between different speaker counts while for 45 seconds the separation becomes visible. Unfortunately, the ZCR value domain for longer analysis durations is different from one mixture to another even for identical speaker counts. In other words, just as in the case of signal power, the ZCR depends on the specific speakers participating in the mixture.

The valuable information we can extract from figure 4.7 is that unlike the signal power, the ZCR decreases with number of competing speakers, when using long analysis intervals. In the case where these two metrics are to be used together we could use the inversed ZCR parameter.

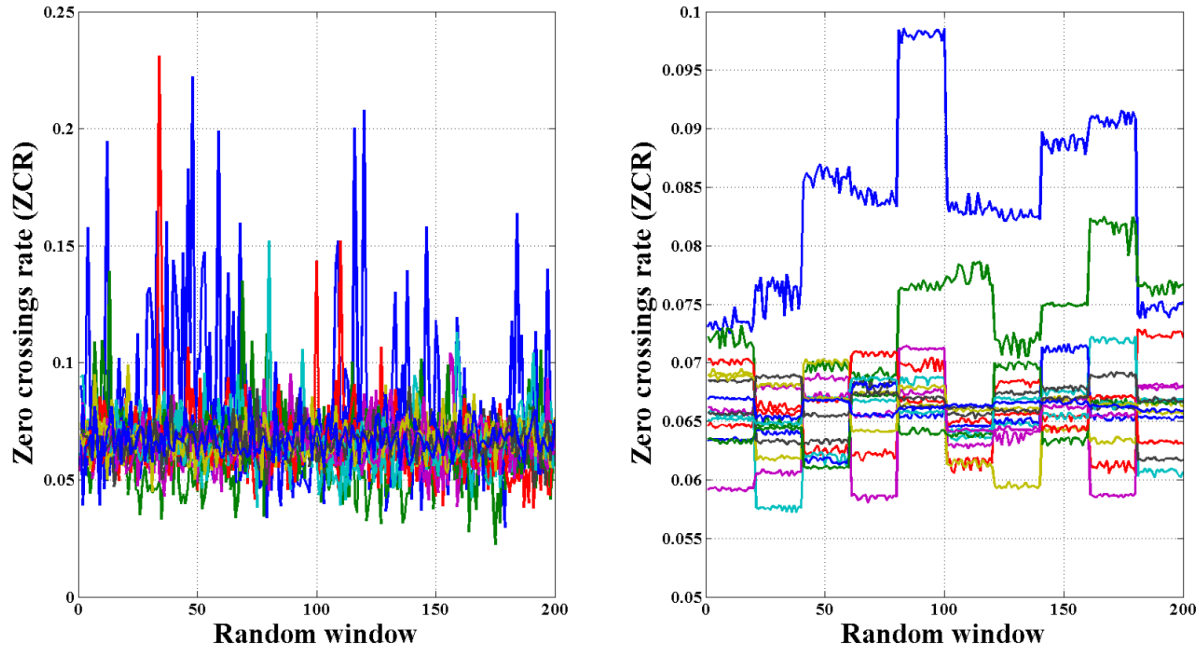

Figure 4.7 – Zero crossings rate (ZCR) separation power

Both the ZCR and the power level for a given window are primitive metrics meaning that they do not contain sufficient information that can be used to detect simultaneous speakers. We move forward by analyzing more complex metrics that can keep information about the time or frequency evolution of a speech mixture.

4.3.1.3 Metrics extraction from power spectral density

Power spectral density (PSD) is a metric expected to modify if the speech signal is produced simultaneously by multiple speakers. The PSD of a given signal can be computed using (4.5).

In the case of a single speaker, when computing the frequency spectrum we will observe a set of peak values for certain ranges of frequencies. These are called formants and the frequency ranges where they appear vary from one speaker to another. This is why we expect that as the number of sources in the mixture grows, the frequency spectrum will become denser, showing formants on more frequency ranges compared to a single speaker source.

$$PSD_{xx}(\omega) = \frac{(\Delta t)^2}{T}\left|\sum_{n=1}^{N} x(n)\bullet e^{-i\omega n}\right|^2 \tag{4.5}$$

There are several methods for estimating the PSD. We used the Welch method, firstly introduced in [Welch, 1967]. This approach is used widely for estimating the signal power associated to different frequency bins and has been adopted because it was demonstrated to reduce noise in the power spectra that it estimates. When the PSD is computed with the Welch method, the signal is divided into frames that overlap. Each segment is windowed, typically using a Hamming window. The next step is to compute the discrete Fourier Transform for each of the segments and average the individual transforms to produce the PSD series. This is however an estimation of the PSD but it is one of the most used methods and has reasonable computational complexity.

The estimation of the PSD produces a set of values, equal to NFFT/2. We will look into visualizing multi-dimensional data into section 4.4 but we also want to investigate if there is a

method for extracting a single value metric from the PSD series. We intend to produce similar charts to figure 4.7 to determine if the PSD estimation can create a metric with more power of classifying recordings with different counts of simultaneous speakers. We used two approaches for extracting the single metric: the mean and the standard deviation of the PSD.

Figure 4.8 reflects the classification power of the proposed methods for 0.5 and 45 seconds analysis duration. The left side of the chart is computed using the mean of the PSD while the right side of the chart is computed using the standard deviation of the PSD. Figure 4.8 also shows similar classification potential to the power level method but appears to reveal slightly less variability to the sources that compose the mixture. We can observe that the line marking the single speaker recording (bottom blue line) is never above the line marking the mixture between 2 sources (bottom green line), unlike the cases for ZCR and power level. Unfortunately, for 0.5 timeframes, we can visually observe that the classification potential of the method is extremely poor, close to what ZCR and power level.

To conclude about PSD estimation via Welch method, we can say that the increasing trend with the number of competing speakers applies also for the mean and standard deviation of the PSD. We also observed slightly better tolerance to the single speaker's sources specifics' compared to power level and ZCR.

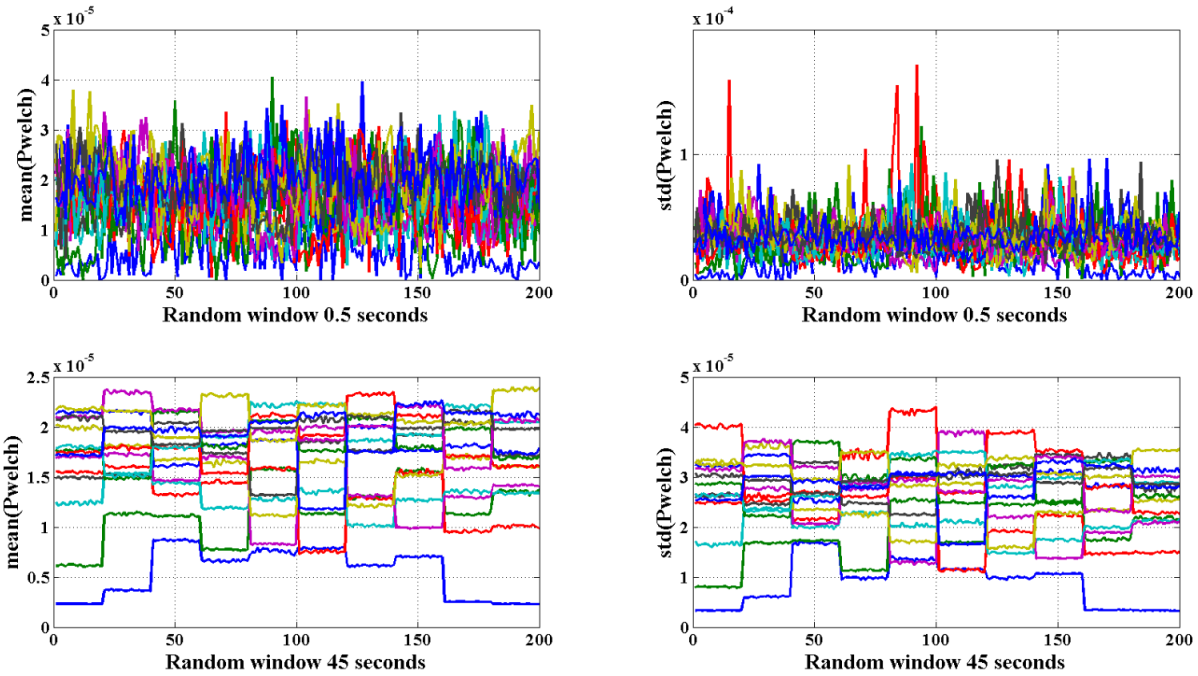

Figure 4.8 – Classification potential of Welch estimation of PSD

4.3.1.4 Estimation based on Shannon Entropy

In a speech mixture we can firmly state that as the number of competing speakers increases, so is the information being carried. In order to determine the number of sources we need a metric that can quantify the amount of information transmitted by the signal. This leads our thoughts to Information Theory. The Shannon Entropy is the exact definition for the expected average value carried by a message. In our case, a message can be a signal frame. This measure was used in speech analysis research by [Renevey, 2001] and [Nilsson, 2006]. The Shannon Entropy can be computed using (4.6), where p_i is the expected probability of the i-th symbol.

$$E(X) = -\sum_{i=1}^{N} p_i \bullet \log_2(p_i) \tag{4.6}$$

There are a few reasons for which (4.6) formula cannot be used directly on speech signals. One problem is the nature of the samples. They are floating point values – in many cases represented with double precision – and it is generally difficult to find two identical. This means that the Entropy will always be constant. The other problem is computational speed. Speech signals have tens of thousands of samples if analyzed in the time domain and thousands of samples if analyzed as a spectrum. Computing the entropy has $O(N^2)$ complexity and therefore a long input set would be prohibitive.

The solution for overcoming these limitations is splitting the input signal into bins. This has $O(N)$ computational complexity and the computation of entropy on the resulted bins has also linear complexity. This transforms the symbol space of the message into an alphabet with as many symbols as targeted bins but stands as a viable solution for computing the Entropy. [Renevey, 2001] divided the spectrum of each frame into bins therefore computing the entropy on the signal's spectrum. [Nilsson, 2006] studies multiple solutions in both time and frequency. For our experiment we used 20 bins when computing Entropy in time domain and 8 bins when using the spectrum as an input parameter. For the spectrum approach we used the PSD estimated with the Welch method. Figure 4.9 showcases the classification potential of this metric.

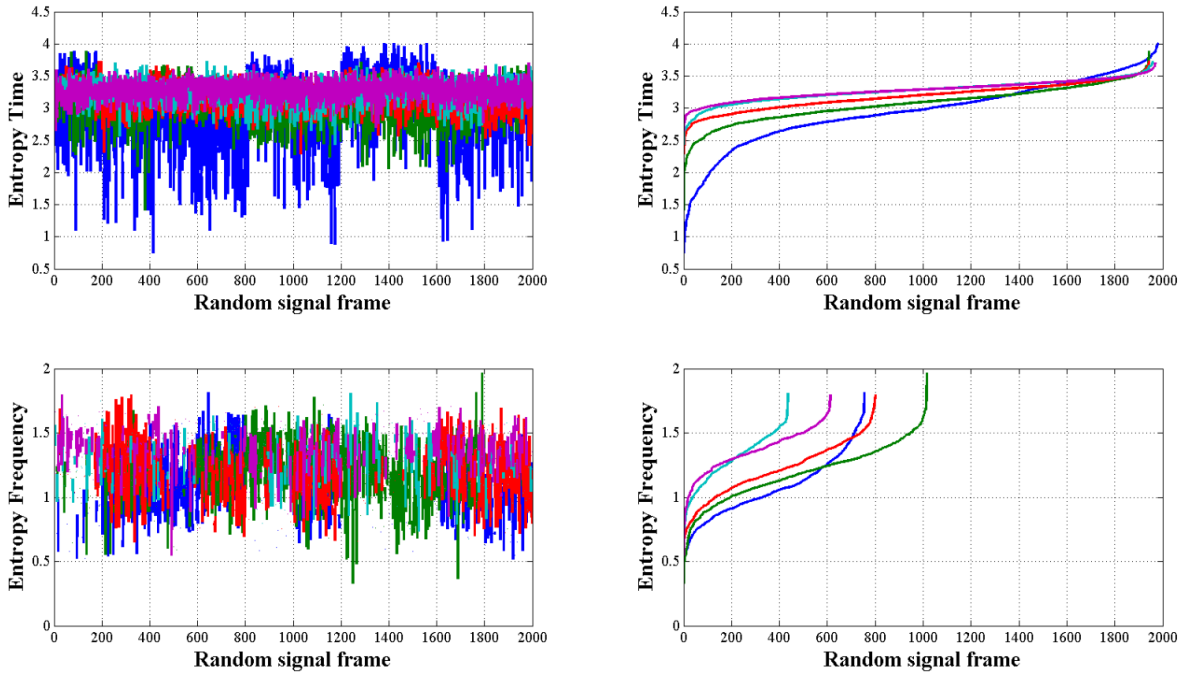

Figure 4.9 – Entropy for randomly taken 1 second frames

As in the case of previously used metrics we cannot say that the classification potential of Entropy is impressive. Compared to previous tentative of identifying a classification criterion where we used 0.5 seconds and 45 seconds frames, in this experiment we used a 1 second window.

We observe a slight increase in Entropy correlated with the increase in the number of speakers but the classification is poor. The right side of figure 4.9 represents the sorted left values of the same figure. We now clearly see the increasing Entropy trend but we also realize

the impossibility of creating bands that would statistically separate the series associated to different speaker counts.

What is more concerning is that even though we used only 8 bins for computing the frequency based Entropy, we get almost half of the time infinite values, which means that at least one bin does not contain values. This might mean that either the number of bins is too large; either the number of FFT points (1024) is small. However, decreasing the number of bins would mean decreasing the alphabet size and consequently the ability of the method to quantify the amount of carried information. On the other hand, increasing FFT points would definitely increase computational complexity and is not guaranteed to solve the problem. Nevertheless, using frequency domain Entropy provides even worse separation compared to using time domain.

To conclude, Entropy at most can contribute among other metrics at associating a recording with a number of simultaneous speakers. Up until now, neither one dimensional metric was not able to provide efficient classification; therefore we must necessarily consider fusion of metrics.

4.3.2 Multi-dimensional feature extraction

The metrics described in paragraph 4.3.1 are single values. As stated earlier they cannot hold too much information about the specifics of multi speaker recordings. Therefore, in the following paragraph we analyze several feature extraction methods that produce a feature vector rather than a single value, in hope we can highlight better classification for the speech mixtures.

We observed that in paragraph 4.3.1 the classification of mixtures with low speaker count appears to be easier. We can observe that for the long analysis durations, the metrics computed for random timeframes appear much more separated for low speaker counts. Also, in real use-case scenarios the number of competing speakers participating actively at a conversation is generally no more than 5. In future sections the thesis will also address the number of competing speakers that can be perceived by a human listener and the presented findings will strengthen the statement that we should focus more on mixtures with speaker counts lower than 5. Therefore, when discussing the feature vector extraction methods proposed in the following paragraph, we will select only up until 4 competing speakers. This will also be beneficial for visualizing the separation potential of each extraction method.

Another limitation we want to address is the analysis duration – or timeframe length. When studying the classification potential of single valued feature extractors we considered extreme frame sizes of 0.5 seconds and 45 seconds. Being able to give a response over the number of competing speakers after 45 seconds of speech has limited practical value. We can think only at automatic data classification for offline recordings. On the other side, 0.5 seconds of speech might contain silence periods and such a timeframe is difficult to be marked accurately with the number of active speakers. We will therefore target timeframes between 2 seconds and 10 seconds in length, trying to lower the bound as much as possible. This will confine more relevance to the speaker count estimation method.

4.3.2.1 Metrics extraction from power spectral density

In the previous section we studied the classification potential of extracting the mean and standard deviation from a mixture's power spectral density computed via the Welch method. We came to the conclusion that the method provides some information about active speakers count but only on long analysis periods. Also, the problem with the mean and standard deviation extraction is that it has the potential of loosing valuable information that can be of use to a smarter classification method.

Figure 4.10 highlights the Welch PSD estimates for random windows of 10 seconds for mixtures from one to 4 speakers. We selected the frequency bins from 80 to 1750 Hz. On the top left corner we can observe the spectrum associated with 1 speaker, on the top right image the PSD is produced by 2 speaker mixtures while on the bottom right image we analyzed a 4 source mixture. We can admit that a pattern that would help classifying the input sets is not observable with the naked eye. Although we hoped we might see a particularity for each speaker count this is not the case.

We studied if this dataset groups in clusters depending on the number of speakers in the mix. To accomplish that we ran a basic k-means clustering algorithm and for each Welch PSD estimates set we obtained the cluster index of the set. Considering that there are 4 data categories we expected that the single source recordings group to cluster 1, two source recordings to cluster 2 and so on. We have 200 samples for each category and we averaged the cluster indexes for each category. We obtained the following averaged array: [1.8, 2.1, 2.6, 3.1]. This means that the recordings with one speaker cluster more to the second cluster than to the first and the recordings with 4 simultaneous sources cluster more to the third cluster. These results show that Welch method computed PSD is not an option for separating speech mixtures into classes, if used as the only option for feature extraction.

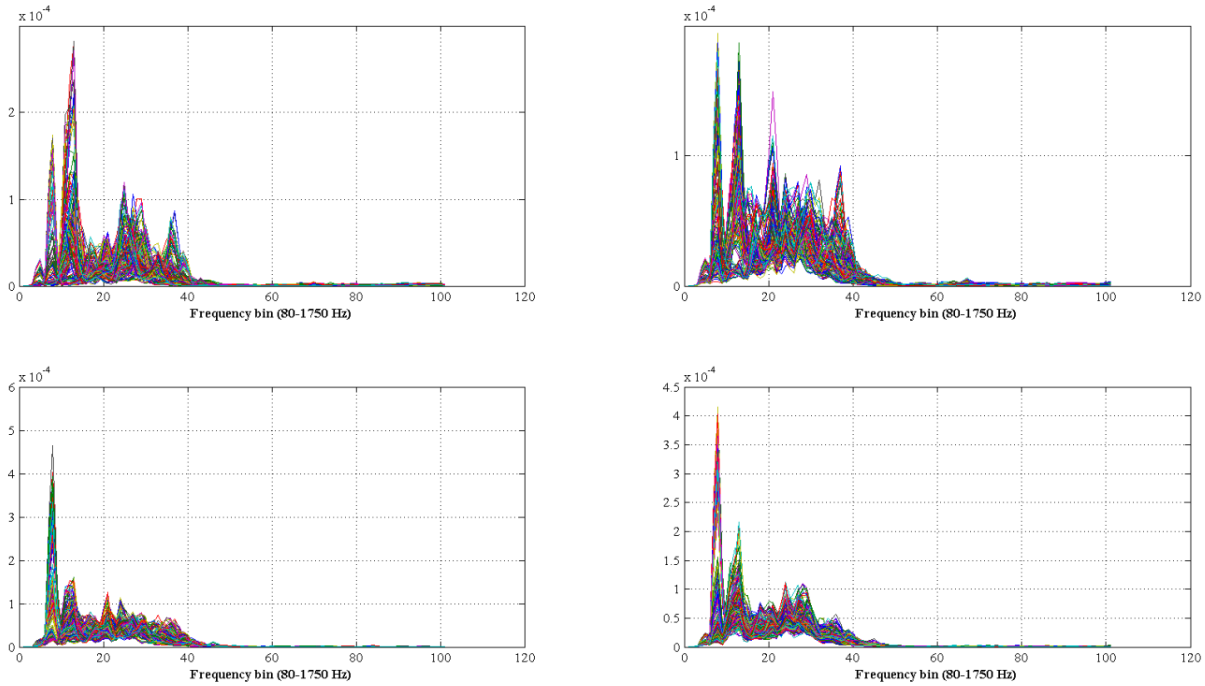

Figure 4.10 – Frequency bins for Welch PSD for 1 to 4 speakers

4.3.2.2 Using MFCC as feature vectors

All the previous attempts for extracting feature vectors from the speech mixtures did not highlight a feature type that would alone provide the necessary classification potential.

Let us investigate using MFCC as feature vectors. [Sayoud, 2010] states that the 7th MFCC can be used to indicate the number of competing speakers in a timeframe. In figure 4.11 we used 4 speech mixtures with increasing count from one to 4 simultaneous sources. We computed the MFCC set for the 4 mixtures considering a 0.5 seconds window, a 0.25 seconds overlap between windows and a 16 kHz sampling frequency. The number of computed coefficients was 12 adding an energy derived coefficient. We sorted the values of the coefficients so, therefore on the *x* axis we have a random coefficient from the MFCC set while on the *y* axis we have the value of the coefficient.

We can observe that the 7th MFCC has some separation power but there is still variation to the specific speakers participating in the mixture. The investigation continued by summing all MFCC values to produce a feature but the value domain for frames with different speaker count continued to overlap. We can conclude that the MFCC set does not carry by itself enough information to ensure a proper classification of speech mixtures.

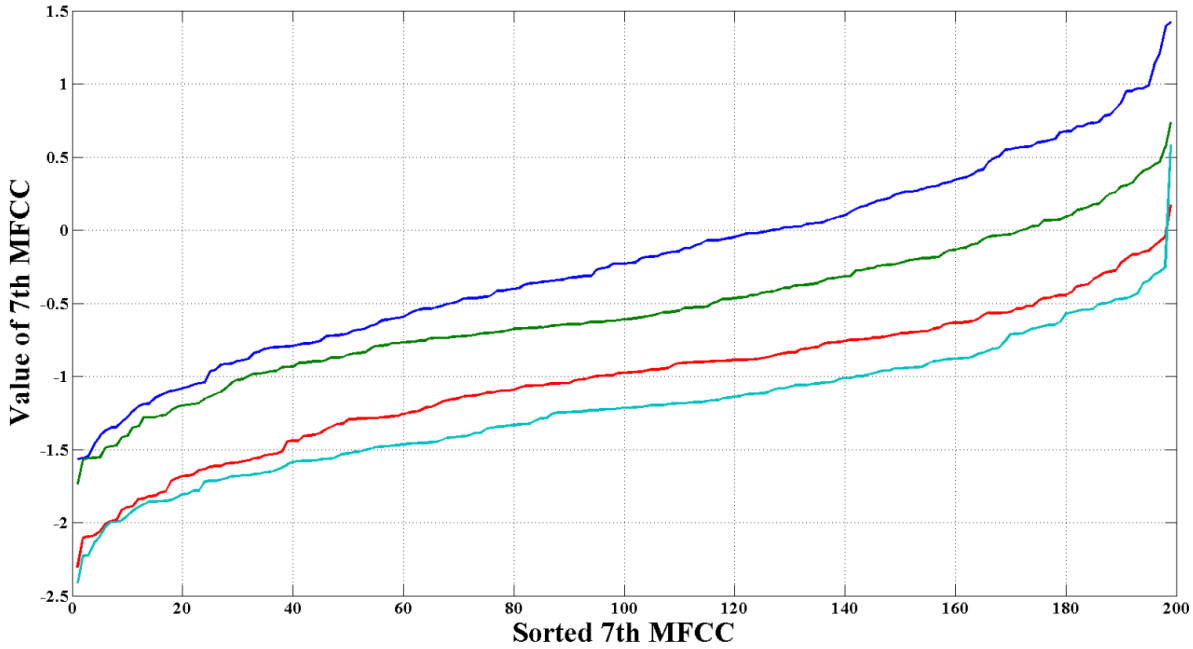

Figure 4.11 – Classification potential of 7th MFCC

4.3.2.3 Using amplitude bins as feature vectors

This method divides the maximum amplitude range of the signal into several bins and counts the number of samples in each bin. It is a fairly straight-forward method with O(N) complexity. We expect that this metric might hold valuable information especially for accurate detection of the lower competing speaker count timeframes.

Of course, this method is vulnerable to mix periods where one or multiple speakers are halting their speech but this is an issue for all other mentioned feature extraction methods. To reduce this issue we could select a longer analysis frame to make sure to capture enough simultaneous speech from all existing sources.

Because the method can generate numerically high values we divide the count of samples in each bin by the total number of samples of the analysis window. The approach can be described by (4.7) set of formulae.

$$\begin{aligned} D &= \left|\max(x_m) - \min(x_m)\right| \\ \Delta D &= \frac{D}{Bins} \\ N_i &= \frac{\sum_{i=1}^{N}((i-1)\Delta D \le x_m(i) < i\Delta D)}{N} \end{aligned} \tag{4.7}$$

As described in section 4.2, a normalization step was performed when mixing the sources; therefore, we should expect that the D variable in the above formula should be the same for all mixtures.

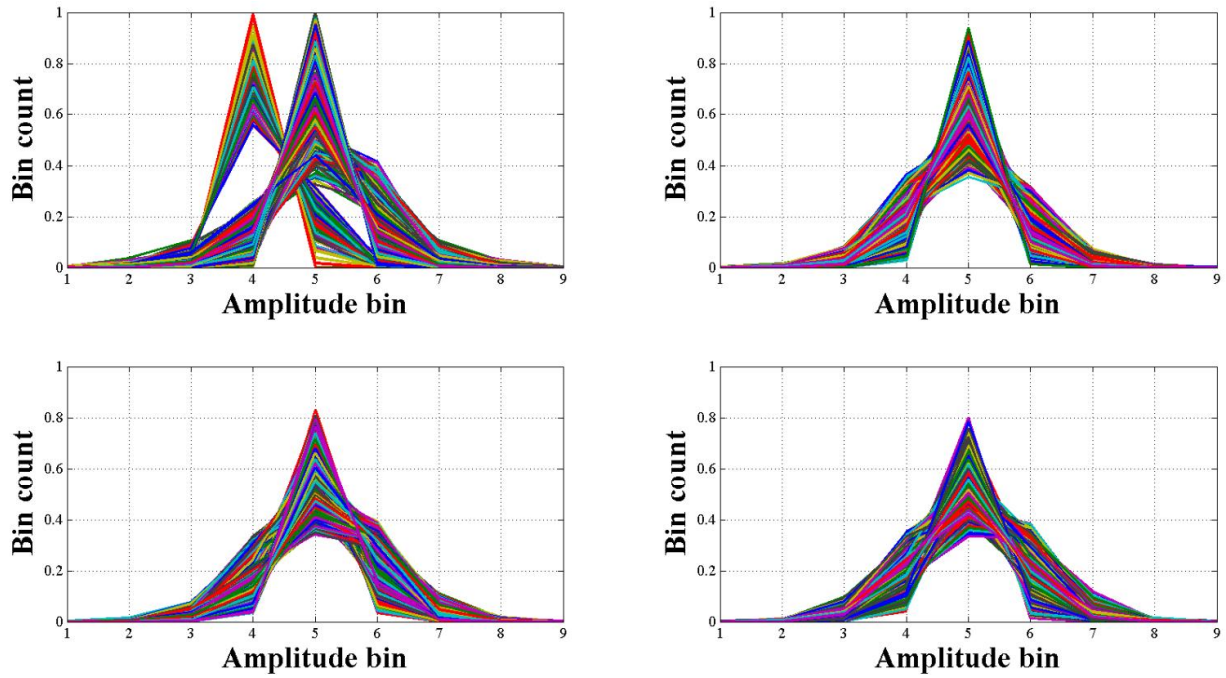

Figure 4.12 – Amplitude bins for 2000 randomly picked 1 second frames

Figure 4.12 is analogue to figure 4.9 except instead of PSD coefficients we display the amplitude bins for 2000 randomly picked frames of 1 second. The top charts are from left to right corresponding to one and two simultaneous speakers. We can observe that the silence periods present much more often when only one speaker is talking, are present as a spike in amplitude bin 4 in the top left chart. Besides this minor observation, we observe similar bin distribution for two, three and four competing speaker counts. Therefore, this feature set has value only for indicating that one or more speakers are talking concurrently.

4.4 USAGE OF SINGLE SPEAKER REFERENCES

All the feature extraction methods that we have tested previously describe the audio signal but not knowing how it should look like. The problem statement is the following: given a set of values that characterize an audio signal, we need to estimate the number of simultaneous speakers using the provided values.

We think that the problem can be rephrased using the following statement: **how different is an audio signal with competing speakers than a set of references?** This is one of the key phrases of this work. Solving the problem assumes that we have a set of references associated to each speaker count and a target reference will statistically bind more to the set of references with identical speaker counts. This method adds more information to various classification methods.

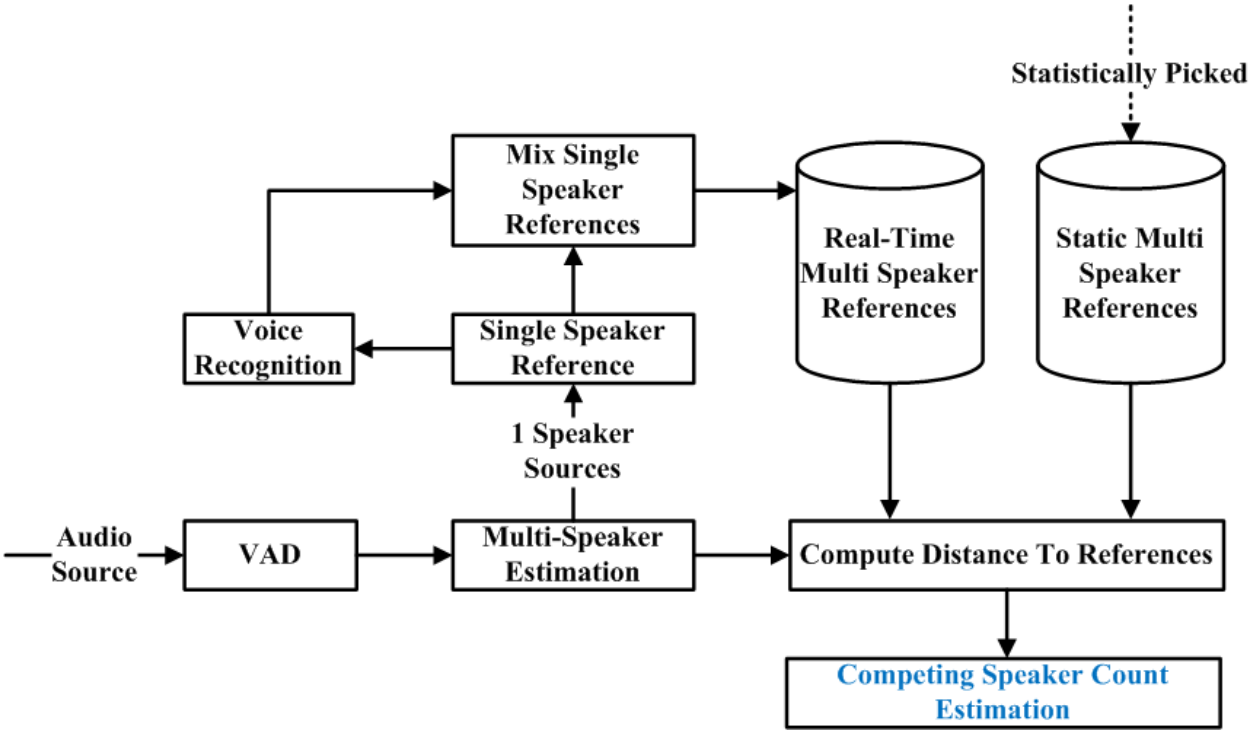

Figure 4.13 – Reference selection proposal

The interesting part is that the reference set can either be built statistically, either computed at runtime. This relies on the fact that the methods for estimating speaker counts have their best accuracy at detecting one speaker. Figure 4.13 highlights the 2 proposed flows for creating the reference set and their usage for determining the competing speaker count.

The approach is rather simple. After the voice activity detector marks signal frames as speech, a multi-speaker estimator is used to select only the frames associated to a single speaker. The estimator has to be tuned for this task but from previous experiments we are confident that the best detection is for one speaker. The next step is to detect if the reference we picked is associated to an existing speaker. For this a voice recognition step is necessary.

Once the single speaker references are captured, the system can mix them to produce references with different speaker counts. These will complete the set of references with the already existing statically picked multi-speaker recordings.

The final steps are to compute the distance of the new recording to each of the existing references and then based on a voting mechanism the estimated number of simultaneous speakers will be computed. The proposals for computing the differences between a target recording and a set of references will be discussed in the following paragraph.

This proposal is indeed a very computationally expensive solution. However, estimating competing speaker count task is not trivial therefore it probably needs a complex solution. The main advantage is that the proposed solution can adapt to the acoustic environment and collect references from the speakers participating to the discussion.

4.5 STATISTICAL AND TRAIN BASED CLASSIFICATION OF MIXTURES

In order for the system to provide accurate classification of speaker counts, it is clear after judging the separation potential of the above features that we need to fuse multiple feature extraction methods. Another strategy that would boost the accuracy of the method is using multiple references associated to different speaker counts. By combining these strategies we can increase the adoption potential for the method of classifying window frames into classes determined by the number of competing speakers.

4.5.1 Feature set selection

Let us first review shortly all the feature extraction methodologies that we have discussed in the previous paragraphs:

- **Signal power** – It clearly shows an increasing trend with the number of competing speakers participating to the mix. However, used alone it does not provide sufficient separation. The advantage is that this increase is linear, therefore comparing to values from a reference and a test frame would show differences following a linear trend with the number of speakers.
- **Zero crossings rate** – This metric has the same properties as the signal power. The only difference is that its value range is usually within *[0.5 0.99]* interval.
- **Power spectral density mean and variance** – The two features also maintain the increasing trend with speaker count but due to the fact that we extract the average and standard deviation we lose information that can be used for classification. Also, the increase in their value with speaker number does not necessarily follow a linear trend therefore it is not trivial to compare values produced using 2 different recordings.
- **Shannon Entropy in time and frequency** – This metric demonstrated some classification potential but not impressive. In addition, because we are dealing with floating point values, it is difficult to associate an occurrence probability to each possible value. To solve this we decided to split the samples into bins but we observed frequent cases where the number of bins was too high and we got infinite values because some symbols had 0 occurrence probability.
- **MFCC values** – We observed a reasonable classification potential only for the 7^{th} MFCC value. The only drawback for their usage is that computing the MFCC demands some processing power and the information brought by the 7^{th} coefficient can be also found in other metrics that can be computed faster.
- **Amplitude bins –** This feature set demonstrated a weak separation potential only for classifying recordings into single and multi-speaker classes. However, by having in mind the flow proposed in figure 4.13 we might continue to analyze this metric as it can be computed without complicating the setup too much.

Considering the fact that we are also interested in comparing feature sets between them we need to introduce several distance metrics. We can use them mainly to operate on power spectral density sets. We analyze using three distance computation methodologies: the **COSH** spectral distance, the **Itakura spectral distance** and the **Itakura-Saito spectral distance**.

The Itakura-Saito distance was the first one introduced in [Itakura, 1970]. This is a pseudo-metric in the sense that it is not symmetric and it measures the similarity between a

reference spectrum and an estimated one. The computation formula can be analyzed in (4.8). A derivation from Itakura-Saito distance is Itakura distance and it was introduce in [Itakura, 1975] and is stated to characterize the spectral density of a given recording. A symmetrical metric is the COSH spectral distance, which is also derived from the two discussed above and is discussed in [Gray, 1976]. The Itakura distance is defined in (4.9) while the COSH distance is defined in (4.10).

$$D_{IS} = \frac{1}{N}\sum_{m=1}^{N}\frac{P(w_m)}{\hat{P}(w_m)} - \log\frac{P(w_m)}{\hat{P}(w_m)} - 1 \tag{4.8}$$

$$D_{I} = \frac{1}{N}\sum_{m=1}^{N} e^{\frac{P(w_m)}{\hat{P}(w_m)}} - \frac{P(w_m)}{\hat{P}(w_m)} - 1 \tag{4.9}$$

$$D_{\cosh} = \frac{1}{2N}\sum_{m=1}^{N}\left[\frac{P(w_m)}{\hat{P}(w_m)} - \log\frac{P(w_m)}{\hat{P}(w_m)} + \frac{P(w_m)}{\hat{P}(w_m)} - \log\frac{P(w_m)}{\hat{P}(w_m)} - 2\right] \tag{4.10}$$

For the following experiments presented in chapter 4.5, we selected only a subset of the discussed metrics to form the feature vectors. Below are the selected features, considering a sampling frequency of 16000 Hz.

- Signal power
- Zero crossings rate
- The active level of speech according to the ITU-T P.56 standard as described in the [ITU, 2012] reference
- Entire estimated power spectral density – This was estimated using Welch method and uses 25 millisecond frames with 12.5 milliseconds overlapping. The number of FFT points per window was 8192.
- Various selections from Itakura-Saito, Itakura and COSH distances

4.5.2 Statistical analysis and voting based classification

This method associates a given frame with a number of competing speakers. There is a maximum number of competing speakers that can be detected. We used for our current experiments up to maximum 5 competing speakers. It uses $\boldsymbol{N_{Ref}}$ references for any speaker count. Below we describe the steps of the algorithm.

4.5.2.1 Reference feature extraction

We use a number of 20 reference frames for each speaker count. This provides reasonable coverage of the dataset and also is a reasonable choice from computational complexity perspective. However, in the case of more complex input frames affected by noise, for example, we can increase the number of references, adding also noise affected samples.

The feature extraction step for references is only computed once and the feature vectors are stored as in-database parameters in the system's internals. However, the distance computation step will require that all the feature vectors from the reference set are compared to the feature vectors computed from the input frame. The reference feature vectors can be formally viewed using (4.11), where $\boldsymbol{N_{Ref}}$ is the maximum number of competing speakers.

Each M_i matrix of feature vectors can be viewed as $M_i=(fv_1;fv_{2;\ldots};f_{VR})$, where R is the number of references for each speaker count.

$$M_{REF} = \begin{pmatrix} M_1 \\ M_2 \\ \ldots\ldots \\ M_N \end{pmatrix} \tag{4.11}$$

4.5.2.2 Feature extraction

For each incoming speech frame, the feature vector describing the frame has to be computed, using the feature extraction methodologies stated in the 4.5.1 section, and also defined in (4.12). The individual components can be computed using (4.3), (4.4) for the signal power and zero crossings rate respectively, while for the last 3 components the formulae (4.8), (4.9) and (4.10) can be used. The active speech level is a more complex feature and cannot be expressed in a single formula, but it is defined in [ITU, 2012]. The last three components operate on the estimated power spectral density using the Welch method estimated in [Welch, 1967].

$$fv(x) = \left[P_x, ZCR_x, AL_x, D_I, D_{IS}, D_{COSH}\right] \tag{4.12}$$

4.5.2.3 Distance computation

To estimate the number of competing speakers, we need to first compute a similarity degree to all the references set associated to each possible number of competing speakers. To compute an initial similarity degree we use the geometrical mean of the references, divided to the newly acquired feature vector, as defined in (4.13). We use the (4.13) formula for each of the 6 components of the feature vector.

After (4.13) we obtain a set of geometrical means of each component. This can also be computed into a single value that can be analyzed to see whether it correlates with a number of speakers participating simultaneously in the speech frame. We used (4.14) for that, which leads us to the final estimator of the number of competing speakers, which also is a geometrical mean. N is the number of references used for a specific speaker count.

$$gm_{f_i} = \sqrt{\frac{C_{1_{reference}}}{C_{1_{input}}}\frac{C_{2_{reference}}}{C_{2_{input}}}\cdots\frac{C_{N_{reference}}}{C_{N_{input}}}} \tag{4.13}$$

$$D(\,frame, references) = \sqrt{gm_1 \bullet gm_2 \bullet \cdots gm_N} \tag{4.14}$$

4.5.2.4 Vote based speaker count estimation

At this point we have a single floating point value that gives an estimate of how similar an input frame is to a set of references bound to a certain number of competing speakers. It is a normal choice to consider that the frame was produced by a number of speakers identical to the one that produced the most similar reference set. We have maximum 5 simultaneous speakers and we have plotted in figure 4.14 the average value of the vector described by (4.15) using 500 recordings. It is expected that if the distance is computed for a frame produced by 2 simultaneous speakers, we will see the minimum value of the distance vector in the second position, and so on for different number of competing speakers.

$$D = [D_{f-R_1}, D_{f-R_2}, D_{f-R_3}, D_{f-R_4}, D_{f-R_5}] \tag{4.15}$$

In figure 4.14 we considered for each speaker count a number of 500 frames. We computed the experiment using 1 second frame lengths and we highlighted on each curve the minimum value. We can easily observe that the minimum value corresponds exactly to the number of competing speakers producing the frame.

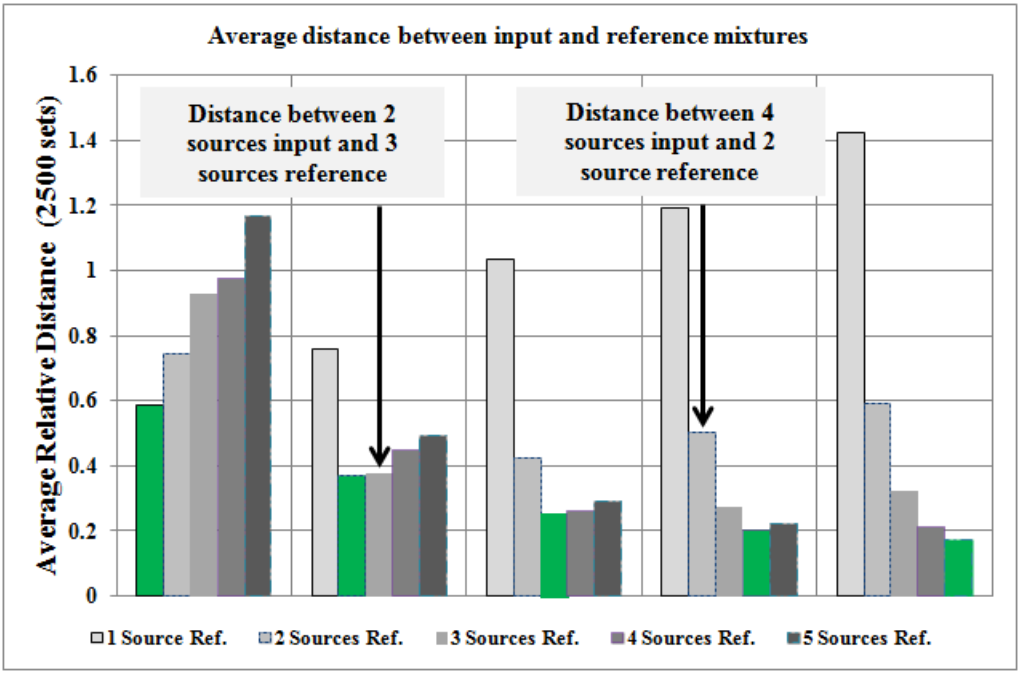

Figure 4.14 – Average distance vector shape for 1 second frames

The above image only highlights us that on average, the features selected for this method of classifying frames to a certain speaker count, can group a set of input frames into clusters where each cluster denotes a certain number of simultaneous speakers. However, the chart does not give information on **how much the clusters overlap**. This is why we need to study the classification performance of the proposed methodology, considering multiple input frame sizes.

4.5.2.5 Performance

Let us now analyze the classification performance of the proposed method. For this experiment we used 500 mixtures for each speaker count. Each mixture was created extracting speeches from a pool of recordings produced by 10 speakers. The references were extracted from the same pool so there *is a mix of unknown and known references* as suggested in figure 4.13. We used 20 references for each speaker count and targeted a maximum of 5 simultaneous speakers.

We define the **False Classification Ratio** metric in (4.16) and we compute this metric for all the competing speaker counts in our experiment, where *N* is 500 in our case.

$$FCR_{speaker} = \frac{\sum_i estimated_i \neq correct_i}{N}$$
$$FCR_{overall} = \frac{\sum_{i=1}^{N} FCR_i}{N} \quad (4.16)$$

Another important aspect to study in this experiment is the variation of the FCR with the frame length. As stated many times in this thesis, it is important to produce a method that would accurately estimate the speaker count using reduced analysis duration. Therefore, we plotted multiple lines for several window lengths. The results can be observed in figure 4.15.

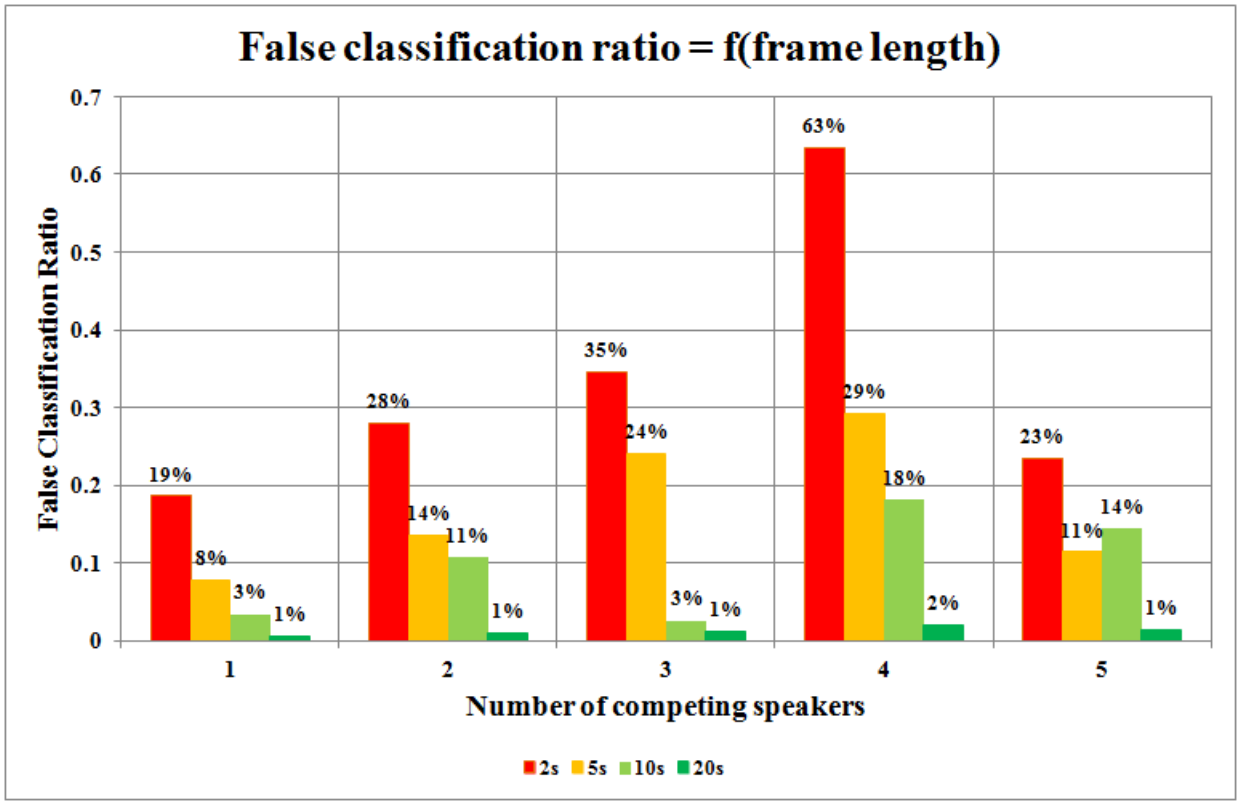

Figure 4.15 – False classification ratio depending on speaker count and frame length

The above picture shows that we can obtain below 2% error when analyzing 20 second timeframes up until 5 simultaneous speakers. However, even the 5 second timeframe analysis does not yield challenging results. We observe about 15% error averaging all the possible speaker counts. This is better error than all the prior work mentioned in section 4.1. Moving to lower analysis durations the error increases almost exponentially. While 2 seconds can still be a duration that can be worked with, showing roughly 35% aggregated classification error, when using a single second analysis frame the error approaches 50% considering the aggregated value.

Another interesting observation that can be made by analyzing the chart in figure 4.15 is that the classification error drops at maximum speaker count. This is caused by the fact that we used the minimum distance to the reference cluster for classification. That is because we observed that as the number of speaker grows, the distance to a small count cluster increases. Therefore, if we would have had clusters for higher speaker counts, we would have observed an increase in the error for 5 competing speakers simply because we can classify the values

into more buckets. Another phrasing of this observation is that the frames collected for 5 speakers are sensible just to under-estimation of the competing speaker count – (*because there are no more classes for higher speaker counts*) – while the latter values are also subject to overestimation.

We will be interested in comparing the performance of this proposed method with that of other approaches. In order to do that we must combine the dimensions we are analyzing into an aggregated performance metric of the method to be able to compare the methods on average. Therefore, we can **average the FCR values for all the speaker counts**. However, since we see clearly that the performance of the method decreases as the analysis duration decreases, we cannot combine performances generated using multiple frame lengths. Table 4.1 highlights the performance of the proposed classification method. The increasing performance trend with the frame length is obvious. We can now use this table to compare it with performances obtained by other classification methods, as discussed in the following sections.

Table 4.1 – False Classification Ratio depending on frame length

Frame length in seconds	Average FCR (%)
0.5 Seconds	54.9
1 Second	47.3
2 Seconds	33.6
5 Seconds	17.2
10 Seconds	9.9
20 Seconds	1.2
45 Seconds	0.7

We also investigated the impact of the reference selection methodology over the performance of the method. For that we selected a 5 second analysis frame and we used the following approaches for selecting the references:

- The references were selected using the exact same mixture recordings. However, the references were randomly selected at different times than the test mixture frames. We expect minimal error for this configuration.
- Both the references and the mixtures were randomly selected from a bucket of recordings produced by 10 speakers. Therefore, when comparing an input frame to the reference set, we get a 50% chance that some of the speakers in the reference mix have also produced the input frame while the other half are unknown speakers. Considering the solution we proposed in figure 4.13 we consider this to be a real usage scenario.
- The references were selected using totally different speakers that the ones who produced the test mixtures. We expect this configuration to yield highest false classification ratio.

Table 4.2 – False Classification Ratio depending on reference selection

5 Second Analysis Frame	5 Speaker pool for references and test mixtures (%)	10 Speaker pool for references and test mixtures (%)	Different pools for references and test mixtures (%)
1 Speaker FCR	1.6	7.8	70
2 Speakers FCR	15.8	13.6	95
3 Speakers FCR	20.4	24	98
4 Speakers FCR	20.8	29.2	45
5 Speakers FCR	9.4	11.4	5

It is beyond any doubt that when the reference mixtures are constructed using a pool of single speaker recordings totally different from the test pool, the classification error increases dramatically. This highlights that the methodology has to be refined if it is to function with fully unknown references.

4.5.2.6 Method summary

Chapter 4.5.2 highlighted that combining feature extraction methods like signal power, zero crossings rate and voice active level computation along with spectral distances can yield accurate classification of input frames into speaker count classes. We used a set of references for each number of competing speakers and each frame was compared to that set of references. In addition to the 3 primary metrics mentioned before, we used power spectrum distance computation to estimate how far is a given spectrum of an input frame to a pre-computed spectrum of a reference.

The set of references were a mix produced by speakers participating to the test set and fully unknown speakers. We highlighted in figure 4.13 that this approach is perfectly possible in a real usage scenario if a pre-processing step that extracts single speaker periods is performed before the estimation. This relies on the fact that estimation of competing speaker count is most accurate for cases when a single speaker is active.

We observed that for a 5 seconds analysis frame, the averaged False Classification Ratio is 17.2%, which is an encouraging result. A 2 seconds frame has 33.6% chances of being tagged as produced by an incorrect number of active speakers while for a 20 second analysis period, the system shows only 1.2% false classification error.

In the following chapter we will investigate if using a learning based approach for classifying the feature set can demonstrate superior accuracy to the method highlighted in this paragraph.

4.5.3 Neural networks based classification

The method of classifying speech mixtures that we discussed above combines several distances computed for each of the components in the feature vector into a single floating point value. This value should describe how similar is a new signal frame with a cluster of frames produced by a certain number of competing speakers. We used the minimum distance as a criterion for associating speaker counts to signal frames. In this chapter we want to investigate whether the classification methodology can be improved by using a neural network.

The combination of relative distances described in (4.12) and (4.13) looses particular effects of the components of the feature vector. With the help of a more complex classifier we

can use the relative distances for all components as an intermediary feature vector that will be tagged with the corresponding speaker count class. In this approach we use an additional feature vector, having N x F components stored in a single line array, where N is the maximum number of speakers and F is the number of components as defined in (4.11). The classifier might observe that a certain component of the feature vector provides better separation power.

Figure 4.16 describes the entire flow for building the train and test sets that can be used for the learning based approach. We start with two disjoint sets of recordings produced by single speakers. The sets do not share a single speaker. Using the two sets we can create a bucket of mixtures with N maximum competing speakers. We then randomly select M reference frames from one set, for each of the N competing speaker counts. At this point we have M x N references for which we extract all feature vectors.

The next step is to generate the train and test database with newly defined feature vectors. Say we want to generate 2500 tests; we select randomly 2500 frames from the test mixtures, keeping in mind that we want equal distribution between different competing speaker counts. For each input frame, we compute the distance of the generated feature vector to the feature vectors. Therefore, we get a new feature vector that has *F components for each of the N speaker numbers*. This feature vector is stored into a database used to train and validate the neural network classifier.

The classifier has to determine a subtle link between the components of the feature vector that would classify the input vectors with superior accuracy.

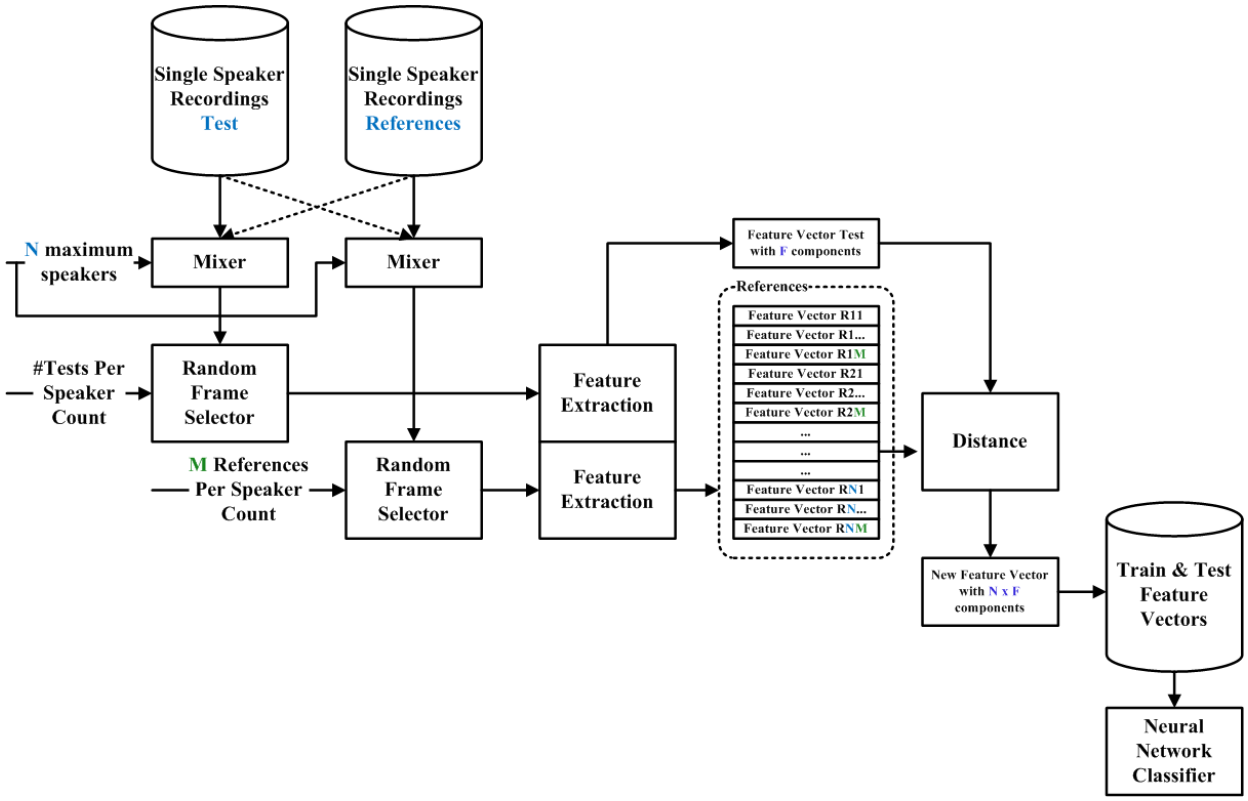

Figure 4.16 – Learning based approach for classifying input frames

The classifier used in this approach was a classic feed-forward neural network with a layer of 50 neurons. We used this number of neurons as we do not expect that the targeted rule of classification is complex and relies mainly on assigning a set of weights to each of the participating components. Also, we estimate that if the neural network has more neurons in its architecture then it is more likely to highly correlate the neuron weights with the specific

dataset being fed for training and we do not want to experience the overtraining effect. Regarding the learning algorithm we used back-propagation algorithm and the Levenberg-Marquardt method for convergence speedup presented in [Marquardt, 1963].

We realize that training a neural network involves fine tuning multiple parameters: gradient, learning rate, momentum and most of all the neural network architecture. This experiment is aimed specifically to showcase if a learning based approach can yield higher classification performance than the statistical analysis classification presented in the 4.5.2. We consider that this goal is achieved if we demonstrate that the reduced size neural network we used in our experiment shows lower False Classification Ratio than the results obtained in figure 4.15.

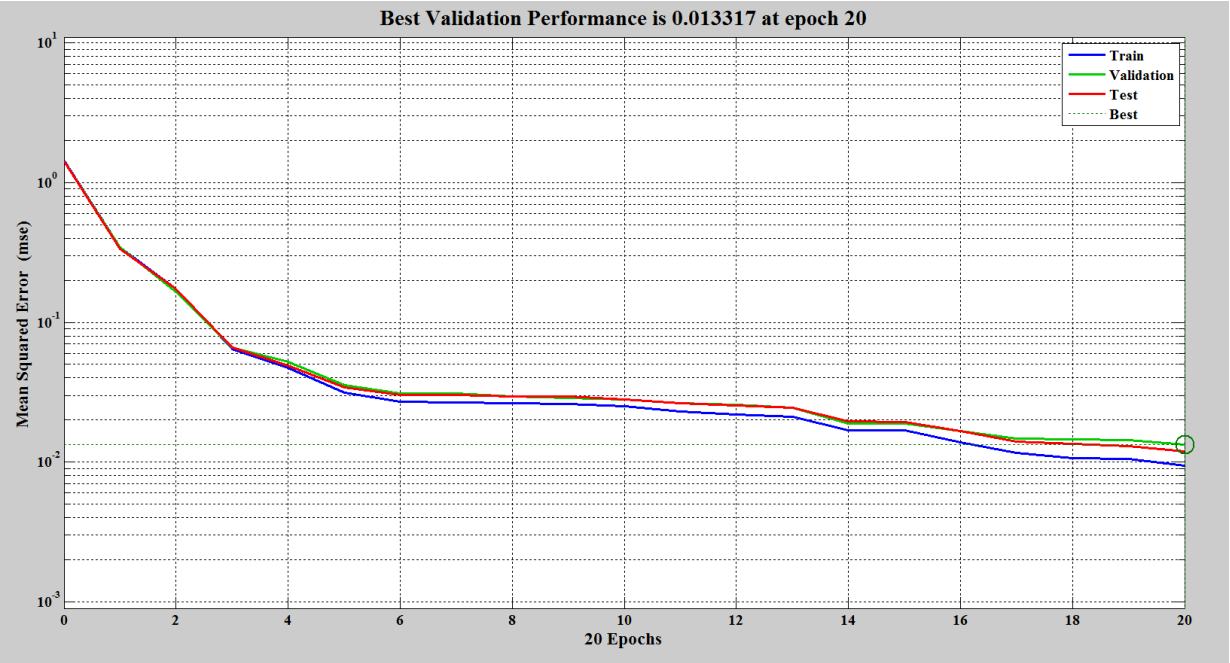

Figure 4.17 – Validation error evolution with the number of training epochs

Table 4.3 – Classification performance gain brought by learning based approach

Error for 5 seconds frame, using the minimal distance voting in 4.5.2 (%)	Error for 5 seconds frame using a FF neural network with 50 neurons (%)
17.2	12

We can observe that the learning based approach can *decrease the classification error up to 30%*, considering a 5 second frame length. Also, in this case, we expect that the classification error would boost when using unknown speakers' references, but the impact might be lower due to the superior performance of the neural network classifier.

4.6 TIME-FREQUENCY ALIGNMENT CLASSIFICATION OF MIXTURES

In this chapter we propose a different approach than in section 4.5. The method still uses single speaker references but due to the computational complexity it does not use references created from multi-speaker recordings. However, it relies on the fact that it is able to determine a single floating point value that would increase linearly with the number of competing speakers. The approach compares an input frame to a single speaker reference (SSR) in both time and frequency.

The reason behind pursuing an additional approach for speaker count classification of input frames is that we want to make the method more robust to fully unknown speakers. We observed by analyzing the above experiments that using completely distant references, the separation power of the method is weak. However, once individual speakers have been identified and recorded as figure 4.13 proposes, the classification performance can encourage the method's adoption into LV-CSR implementations.

4.6.1 Method overview

Figure 4.19 reflects the entire process of determining the similarity degree between a simultaneous multi-speaker recording and a SSR and the mapping to a speaker count.

The first step is to compute a spectrogram for each of the signals. We used a window length of 0.5 seconds with an overlap of 0.125 seconds and 1024 points for the FFT. The sampling frequency was set to 11025 Hz. We tried other feature extraction methods like PLP and MFCC but we discovered that this approaches generate a higher error rate. The reason is that MFCC and PLP are efficient when using narrower signal frames (≤ 0.1 seconds) while our method uses larger windows to better capture the effects generated by competing speakers. The selected window and overlap length were observed to determine performance sweet spots, after running multiple automated experiments.

The next phase is comparing all possible pairs of window spectra computed for the two recordings using a distance metric. We can use the spectral distances defined in (4.8), (4.9) or (4.10) or we can use the Dynamic Time Warping distance.

The computational demands of this method are high. For example, when using 2 recordings of 10 seconds generating 26 windows each, 676 distances are computed in a 2D structure. In order to obtain a single similarity value, we add all the values in the distance matrix. We discovered that this approach generates good accuracy. A method of eliminating distances in the 2D structure would however decrease the computation time of the solution.

The final step is correlating the similarity degree with a previously computed trend. We observed that the trend can be modeled by a monotonic function (in most cases, if high quality SSR are used, the trend is linear). For each speaker count we can compute a set of thresholds on the trend. The number of active speakers results from the closest threshold lower than the distance to the SSR.

The trend and thresholds are obtained statistically, before runtime, by considering multiple SSR and averaging the similarity degrees. If the number of used SSR is higher, the extrapolation capability of the solution is expected to increase. In the case of signals heavily affected by noise, it is possible to compute the distance trend and the thresholds dynamically, during runtime, for environmental adaptation.

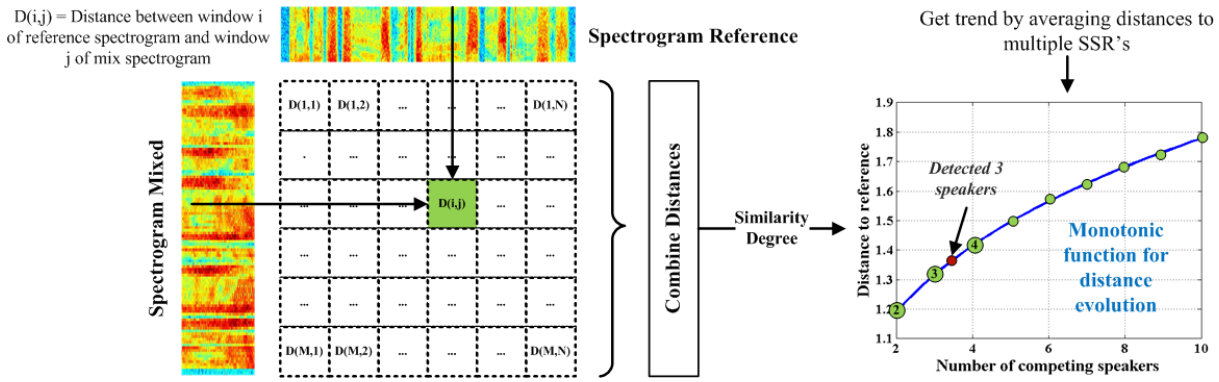

Figure 4.19 – Method for time-spectrum alignment of speech mixtures

The proposed method has the input parameter set described below and gives a single similarity degree between two speech mixtures.

- **Input signals** – Can be any audio signals but the method is relevant just for speech;
- The spectrogram **window and overlap sizes in seconds** – We used a 0.5 second window with 0.125 seconds overlap. This was determined experimentally;
- The number of **FFT points** – We used 1024 points as a reasonable tradeoff between precision and computational demands;
- The **sampling frequency** – We used a 16000 Hz value for all our recordings.

Table 4.4 – Matlab source code for time-spectrum alignment of speech mixtures

```
function distance = get_dtw_spectrogram( ref_frame, in_frame,
    win_seconds, overlap_seconds, n_fft, fs )

m_spectr = abs(spectrogram(in_frame, floor(win_seconds * fs),
    floor(overlap_seconds * fs), n_fft, fs));
m_N      = size(m_spectr);
m_N      = m_N(2);

s_spectr = abs(spectrogram(ref_frame, floor(win_seconds * fs),
    floor(overlap_seconds * fs), n_fft, fs));
s_N      = size(s_spectr);
s_N      = s_N(2);

d_sm = ones(m_N, s_N);
for k = 1 : m_N
    for l = 1 : s_N
        d_sm(k, l) = distance(m_spectr(:, k), s_spectr(:, l));
    end
end
distance = sum(d_sm(:));

end
```

Table 4.4 provides the Matlab code needed to test and analyze the proposed method. The *distance* function can be any type of function that shows the capability of determining similarities between spectra. We used mainly Dynamic Time Warping distance but COSH distance provided in (4.10) can be of use as it is a symmetrical distance.

Currently we use a single approach for combining individual distances between spectrogram windows: we add them. This approach can be studied for improvement by investigating whether there is a key selection of indices that can keep the same amount of information for proper classification of input frames. Such an approach would also decrease the computational requirements of this method.

4.6.2 Optimized "Dynamic time warping"

The most challenging computational task of the proposed method for associating a signal frame with a number of simultaneous active speakers is being able to calculate the DTW distance efficiently. This algorithm is mostly viewed as sequential and has $O(N^2)$ complexity

The main performance drawback is that Matlab is an interpreted language. This means that roughly each instruction executed by the Matlab runtime environment has a correspondent set of native instructions. This makes the code to perform slowly because the machine's CPU will be underutilized. This is caused by the fact that optimal code is not generated by an efficient compiler.

Because we need the DTW routine to perform fast, we decided to implement it natively using C and compile the code optimizing it for the CPU architecture we used to run the measurements. In addition, we used a compiler specifically optimized for the CPU that we used. Nevertheless, we also redesigned the algorithm to be able to fully utilize hardware resources but also yield identical numerical values.

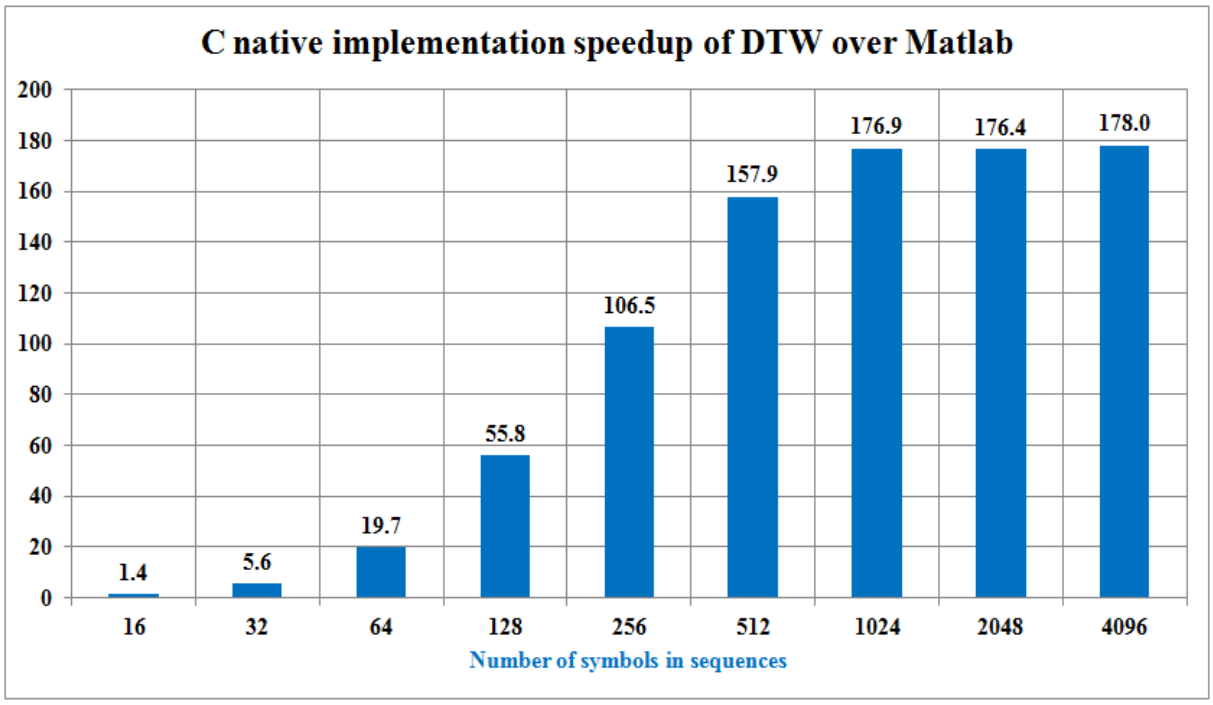

Figure 4.20 – Speedup of C implementation of DTW

The approach was simple. We written the DTW algorithm using C and created a dynamically linked library using the Visual Studio Integrated Development Environment. We then loaded the library using Matlab that provides an interface for doing this. When stating that we redesigned the algorithm to fully utilize computing resources, we used a *waveform* implementation that enabled the usage of Single Instruction Multiple Data (SIMD) instructions and allowed better caching of sequences to be compared.

Figure 4.20 highlights the speedup over the identical DTW code implemented in Matlab. We can observe 150X speedups for sequences of 512 elements as were mostly used in our experiments. We must state that in order to collect the data for a given method configuration we ran experiments that lasted close to two days. Without the 150X speedup these experiments would have approached one year.

4.6.3 Performance comments

For the experiments presented in this chapter, we will use single speaker references that did not participate in the mix and compare classification performance to the case when we are using a combined pool of recordings that have 50% of participating in the mix. We can use maximum 5 single speaker references due to the fact that we had a database created by 15 speakers.

As in the case of previous discussed approaches we will investigate the impact of frame length over estimation accuracy. It is expected that the longer the timeframe the more accurate can the method be. In addition to this experiment we also want to investigate the effect of the number of references over the classification accuracy and also the impact of noise.

The following paragraphs will use the averaged FCR as defined in (4.16), unless stated otherwise. We want to be able to reflect an image over the entire speaker count spectrum and not focus on any outliers that might show up when producing a FCR for each speaker count.

4.6.3.1 Frame length

As in the case of learning and statistics based classifiers that combine multiple single valued features, the length of the analysis frame is important. The longer the analysis duration, the more information about the number of competing speakers is being held.

Table 4.4 shows the differences in the FCR metric for both reference selections cases.

Table 4.4 – Classification error depending on frame length

Frame duration in seconds	FCR references participating to the mixtures (%)	FCR references not participating to the mixtures (%)
0.5 seconds	55	86
1 second	47	81
2 seconds	31	73
3 seconds	22	56
5 seconds	15	47
10 seconds	6	35
25 seconds	1	3

We can easily observe that the classification when using references that have participated to the mixture is far more superior to when using unknown sources. We must also state that this experiment was done using up to 10 simultaneous speakers. However, the frame length investigation tells us that even when using fully unknown speakers as references we have the chance of observing excellent classification for frame lengths higher than 10 seconds.

This was one of the purposes of this new method: to provide superior accuracy when dealing with fully unknown speakers. However, in real use case scenarios, the speakers participating to a conversation do not vary a lot therefore, if the single speaker voices are captured correctly, then the method is expected to show the classification performance associated to the case when single speaker references are participating to the mixtures.

4.6.3.2 Number of references

For this test we will select the reference set that also have a 50% for participating to the mixture. We used this approach because the classification errors are lower and the impact of the number of references can be highlighted easier.

Figure 4.21 shows the results of this test. We plotted the FCR per speaker to also highlight that the error grows with the number of speakers participating to the mix. We considered a 10 second analysis duration.

We can see that while for a single reference the error can go as far as 45%, if we use the entire set of 5 single speaker references, the maximum FCR is roughly 12%. We also expect that this error might shrink if we use more references, but of course that will increase the price paid for computational resources.

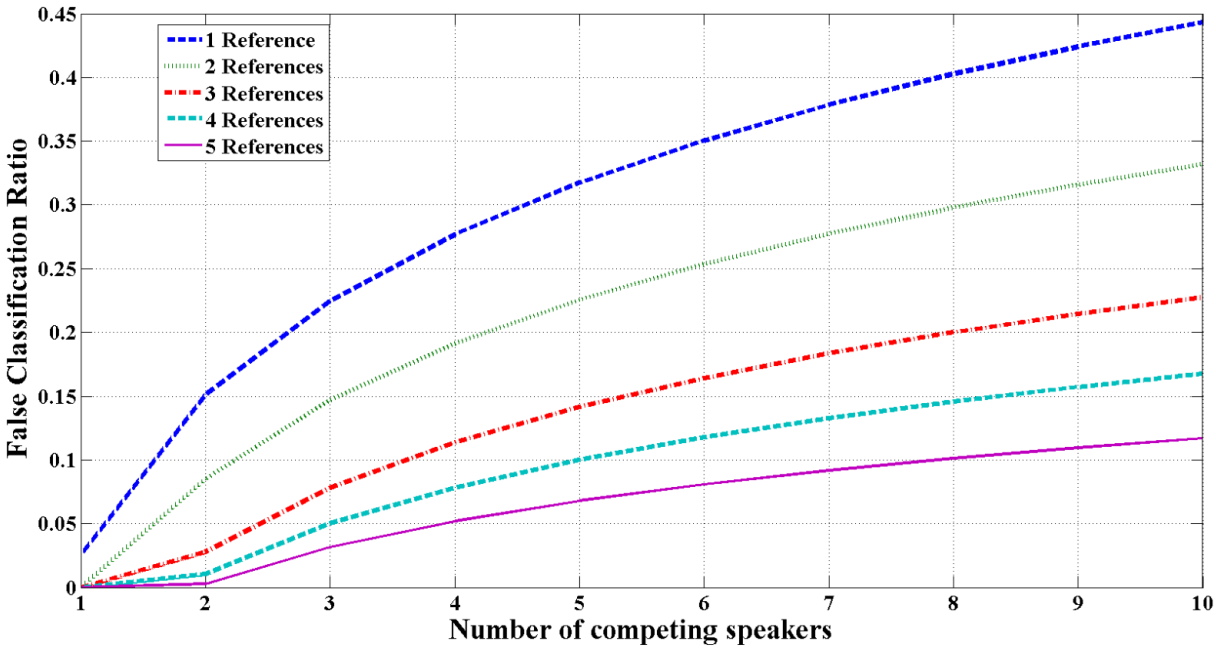

Figure 4.21 – Reference count impact over the FCR

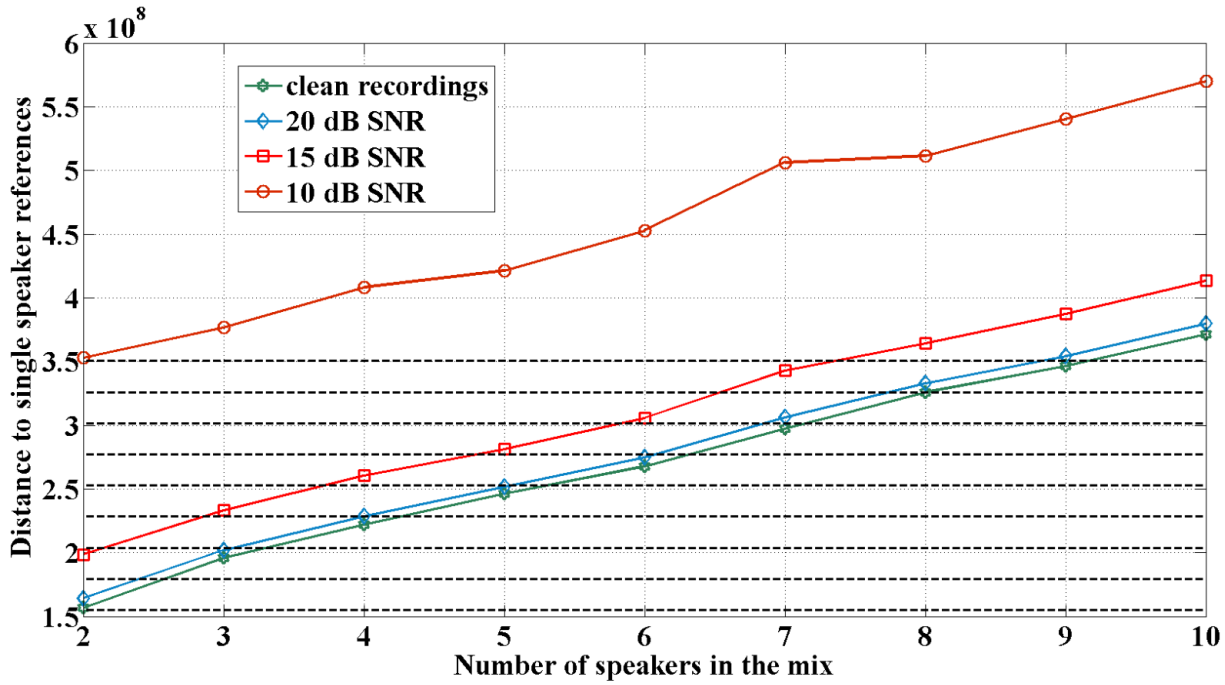

Figure 4.22 – The influence of noise over the speaker count estimator value

4.6.3.3 Noise influence

Noise can also play an important role in the accuracy of the proposed methodology. We want to highlight how it can tamper the FCR. We also used the mixed pool of recordings as in the previous case because we want to isolate the effect of noise only.

Due to the fact that this methodology uses a set of thresholds computed statistically, the risk with noisy recordings is that they would cross the threshold range that was assigned for an identical number of active speakers. In other words, the speaker count estimator value has to stay within the ranges that indicate the correct number of speakers. The noise can shift the estimator value outside the correct interval.

Figure 4.22 highlights that a 20 dB signal to noise ratio recording does not generally shift the distance to the single speaker references outside the intervals indicating the correct speaker count. However, a lower SNR can raise serious challenges for the accuracy of the method. The noise added in these recordings was white Gaussian and because we are aligning spectrograms using DTW, which increased systematically the value of the speaker count estimator.

For example, a value of 1.8×10^8 indicates correctly 3 competing speakers for both clean recordings and mixtures affected by a 20 dB SNR. When the SNR is 15 dB the estimator value grows to 2.6×10^8 and that corresponds to 5 competing speakers. If the SNR is 10 dB due to the estimator value increase the estimated speaker count is highly above 10.

In order to equip our approach with robustness to noise, a set of references affected by noise should be used in order to be able to compute thresholds taking into account the noise effects. Figure 4.22 tells us that the accepted SNR is maximum 20 dB. After this value the estimation performance is highly expected to drop drastically.

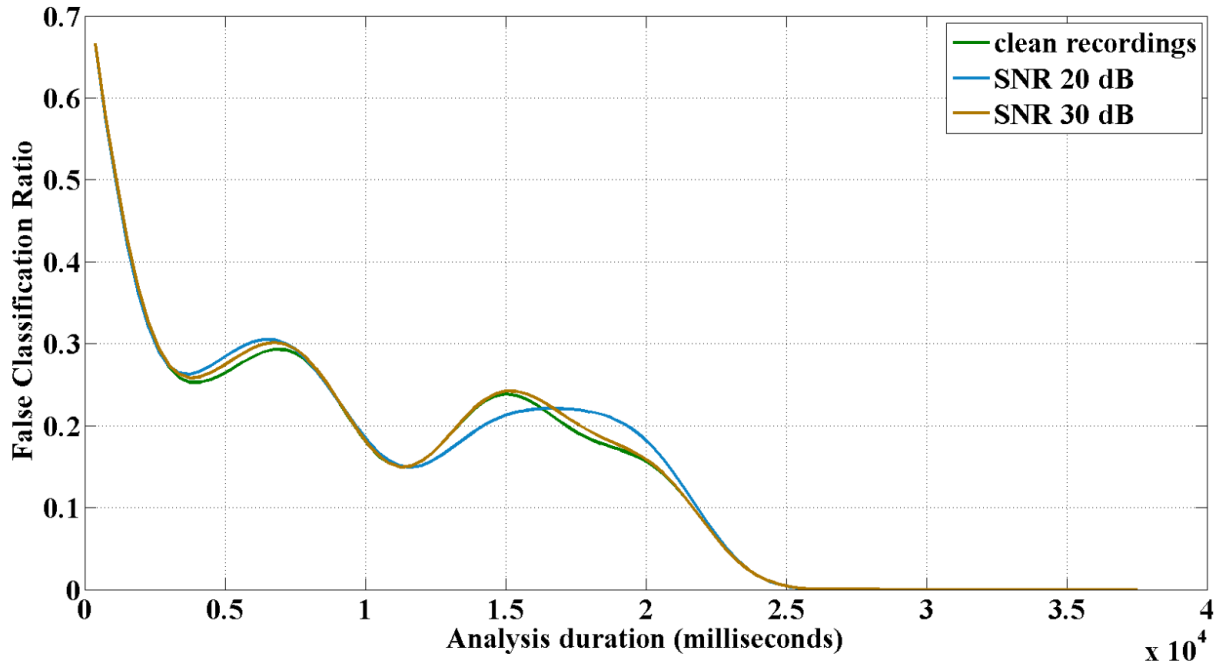

Figure 4.23 – Noise and analysis duration impact on FCR

The final experiment that we did in order to analyze the speaker count classification method is to highlight how the noise influences the frame length for optimal classification ratio. Figure 4.23 showcases how a different analysis frame can affect the FCR, for clean mixtures and mixtures affected by 20 dB and 30 dB SNR. We can easily observe that the noise affected mixtures do not affect the trend produced by clean samples.

4.7 SUMMARY

The entire chapter focused on analyzing various methods for determining the number of competing speakers that are participating in a speech mixture. There were two main approaches discussed:

- **Learning based and statistical classification** – These methods collected features like the signal power, the active level, the number of zero crossings, the entropy, various MFC coefficients, the mean and variance of the power spectral estimation using Welch method and statistically analyzed the trend of the variables with the number of competing speakers. We demonstrated that all the mentioned features increase or decrease according to a monotonic function with the number of speakers but they do not hold sufficient information for pure threshold based classification.

In order to associate a number of competing speakers to a signal frame, we used single and multi-speaker references. For these references we computed the set of metrics enumerated above and compared them with all input frames with unknown speaker counts. We also introduced a set of spectral distance computing methods, especially for estimated power spectra, that were used to obtain the metrics when references were used. The statistical methods combined all differences between metrics in a single estimator that should have indicated the number of competing speakers. This demonstrated a fair degree of accuracy.

To further improve the accuracy of the method we used a learning based approach to classify the frames and associate them with numbers of active speakers. We used a feed-forward neural network, trained using Levenberg-Marquardt approach and back-propagation that was supposed to take as input all the differences between the metrics generated by an unknown frame and the set of references. The experiment demonstrated that neural networks can yield superior accuracy to statistical based processing mostly by being able to select the relevant components from the feature set that can provide the best accuracy.

- **Time-frequency alignment of mixtures with single speaker references** – This methodology aimed to provide a "brute-force" approach for obtaining an estimator over the number of competing speakers. Each unknown speaker count frame was compared in time and frequency using Dynamic Time Warping with a number of single speaker references. This produced a matrix of distances that were combined to produce a single value estimator of the competing speaker count.

For both methods that we discussed in chapter 4, we were interested in performance analysis. We were mainly interested in how long the input signal frame has to be in order to provide enough information for proper classification. We observed that an increased frame length can produce superior classification performance. We also demonstrated that there are methods to lower the necessary frame length by using more references or different features for example.

We were also determined to study the impact of using fully unknown speaker references against the case when a mix of known and unknown references was used. This analysis demonstrated as expected that fully unknown references can decrease the classification performance. However, in real conversation scenarios the number of speakers does not vary greatly and therefore we rather need a methodology for extracting the single speaker voices and use those as references. We proposed a schematic of such a system in this work.

The number of used references was also analyzed to reflect its impact over the classification accuracy. As expected, the more references the classification error is expected to decrease due to a more representative computation of the competing speaker's number indicator.

Lastly we highlighted the impact of noisy recordings over the method's ability to classify input frames. We observed that especially in the case where speaker counts are computed based on thresholds; the noise applied to recordings has the undesired effect of shifting the estimator values outside the associated interval. However, for recordings with SNR higher than 20 dB the effect is greatly reduced allowing the method to keep a good classification accuracy.

5 BLIND SPEECH SOURCE SEPARATION

Blind source separation (BSS) is a broader research study that targets among others, a set of methods designed to deal with speech signals. Researchers in the field of BSS aim to produce algorithms that can be used for separating individual signals from a mixture. Some of the components in the mixture can be undesired (e.g. noise, irrelevant information, etc.) or all identified signals can be of interest. BSS has numerous applications, as beautifully illustrated in [Kokkinakis, 2010]:

- **Acoustics –** *Improving signal quality***:** Cross-talk removal, speech separation, auditory perception, scene analysis, coding, recognition, synthesis and segmentation, psychoacoustics, reverberation, echo and noise suppression and cancellation, signal enhancement, automatic speech recognition (ASR) in reverberant and noisy acoustical settings. Potential uses in mobile telephony, hands-free devices, human-machine interfaces (HMIs), hearing aids, cochlear implants, airport surveillance, automobiles and aircraft cockpit environments. Music processing falls within this area with [Sakuraba, 2003], [Samaragdis, 2003], [Raphael, 2002] and [Vincent, 2004] studying the separation of musical instruments from a melody recording followed by note transcription. Another example is when [Mansour, 2006] describes a method for separating underwater acoustic signals.

- **Biomedical processing** – *Improving medical imaging and biomedical signal quality*: Enhancement, decomposition, artifact removal of electroencephalograms (EEGs), electromyograms (EMGs) and magneto-encephalograms (MEGs). Another example is non-invasive separation of fetal from maternal electrocardiograms (ECGs). [Anemuller, 2003] and [Vaya, 2006] present some applications of using BSS in the field of biomedical processing.

- **Financial applications** – *Analysis of sequences of economical indicators*: currency exchange rates or cash flow data analysis, stock prediction, minimization of investment risks, etc.

- **Geophysics** – Exploration of Earth's crust layers, seismic monitoring,

- **Digital communication** – *With great emphasis on quality improving*: Smart antennae design, direction of arrival (DOA) estimation, adaptive beam-forming, multichannel equalization, multiuser separation. Potential uses in wireless

communication schemes, mobile radio and telephony and satellite telecommunication systems.

- **Statistics** – Factor analysis, data mining, feature extraction and classification, Bayesian modeling, information theory.

- **Image processing** – Image and video analysis, denoising, compression, restoration and reconstruction, computer vision and tomography, visual scene analysis, motion detection and tracking, medical imaging, etc. For example, [Nullizard, 2000] highlight an algorithm for analyzing astronomical images using BSS techniques.

Dealing with speech signals makes BSS a much more difficult task because speech is one of the most difficult signal types to be studied and analyzed. Section 4.1 will deal in detail with the challenges of speech source separation. The main focus of blind speech source separation, later referred as BSPS is to extract a set of high quality signals that can be later used for automated analysis. The most common task that can be exemplified is doing automatic speech recognition simultaneously for multiple speakers in a room. This represents the well known cocktail party problem, phrased by Colin Cherry in 1953.

5.1 CHALLENGES OF SPEECH SOURCE SEPARATION

As stated above, BSPS has to overcome numerous challenges related to the complexity of the speech signal. Some of these challenges can be: the variability of speech signals, spectra overlapping, time-frequency overlapping, reverberation effects, source movement and the underlying effects, noise that affects the signals, the different speech intensities and speeds associated to each speaker. In the following paragraphs we will exemplify each of the mentioned challenges.

Let us start with the most 2 basic and simple to highlight challenges. One is variability of speech signal considering the associated waveform and the other is the overlapping of spectra of speech signals. In figure 5.1, we highlight the waveform associated to 4 consecutive pronunciations of ‘A’ vowel.

We can observe how while the signals resemble they are not identical. This imposes much more complex analysis methods to draw conclusions about the information encoded by the speech signal, and makes it difficult to analyze the signal by solely considering the time domain. Moving to frequency domain we also encounter challenges. Figure 5.2 highlights in the top part 2 spectra associated to speech signals produced by 2 persons. One is a female speaker and speaks the word “father” while the other speaker is a male and speaks the word “mother”. We can see how the 2 spectra overlap. Of course, there is a shift towards lower frequencies for the male speaker but a clear separation is far from being visible. The lower part of the graph shows the spectrum associated to the signal mix of the 2 speakers. We can see that the mix spectrum contains elements from both of the above spectra. The main problem in doing BSPS is that there is no knowledge of individual sound sources composing the mix, so it is extremely difficult to create a method of separating two or more sources by just analyzing the spectra.

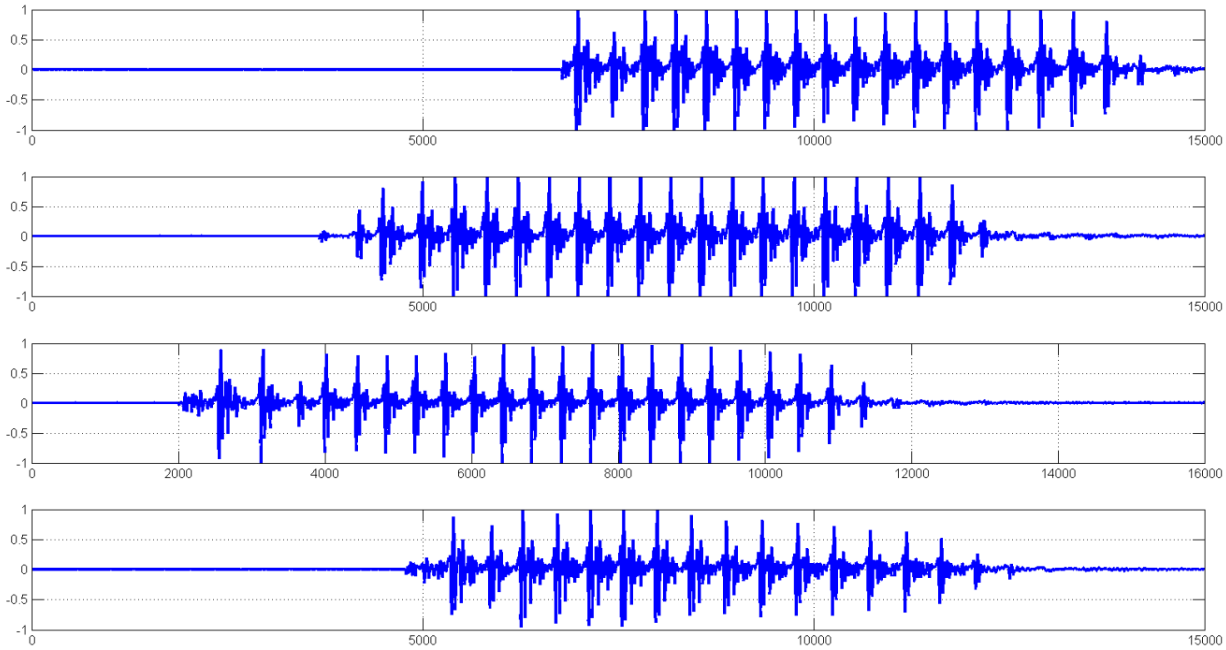

Figure 5.1 – Consecutive pronunciations of ‘A’ vowel

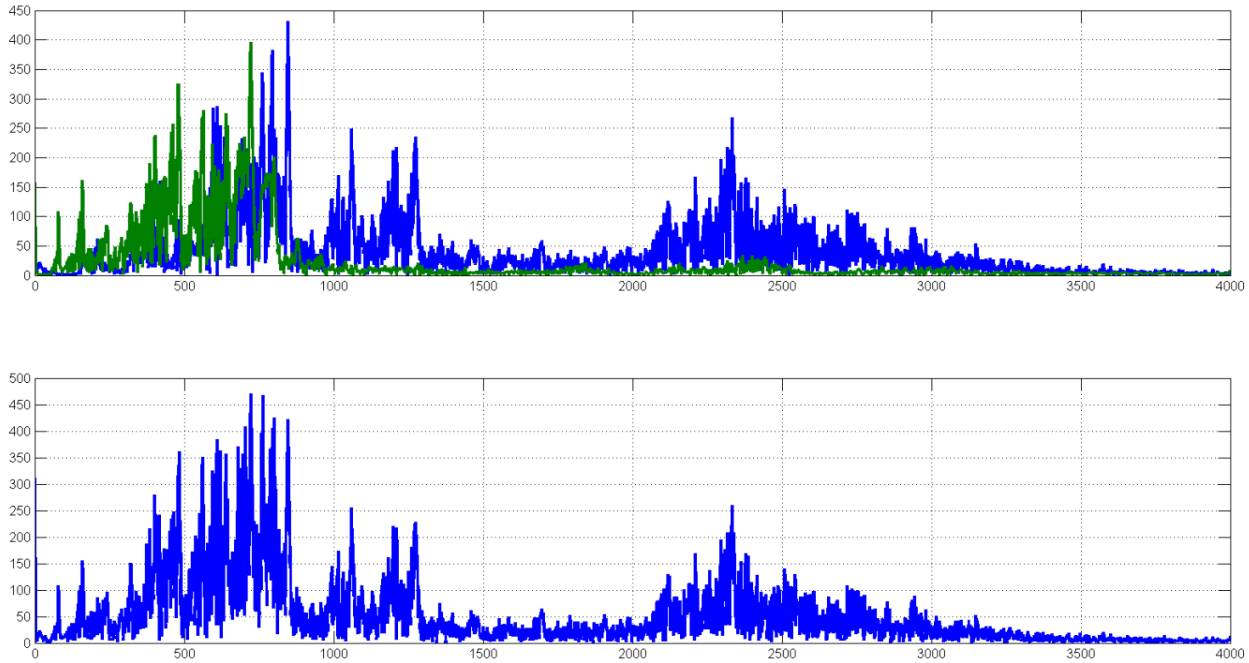

Figure 5.2 – a) Spectra associated to a female spoken ‘father’ (blue) and a male spoken ‘mother’ (green); b) Spectrum associated to the mix of the above signals.

The above figure shows how spectra computed from signals belonging to a female and male speakers overlap. Having two or more persons from the same gender, makes the separation of sources even more undetermined.

Another subtle challenge that needs attention when doing BSPS is reverberation. **Reverberation** can be defined as the phenomenon of having the sound persisting after the moment the sound was produced. This is generally caused by reflections of the sound on the walls or obstacles present in the recording room. The reverberation is usually perceived as an echo. Figure 5.3 shows an example where we have a clean speech signal and a signal affected

by reverberation, in the lower part of the chart. We can see how the green component "arrives" later at the sound receptor and further on the red component arrives having lower amplitude. The easy to observe effect is the increase of signal amplitude in the region between the 2 syllabuses. Reverberation inflicts even more complexity to the speech analysis methods and makes the separation of sound sources even harder because it is often difficult to detect if the analyzed speech is the main component or a reflection of the initial sound. We will discuss in the following paragraphs about methods to overcome some of the discussed limitations.

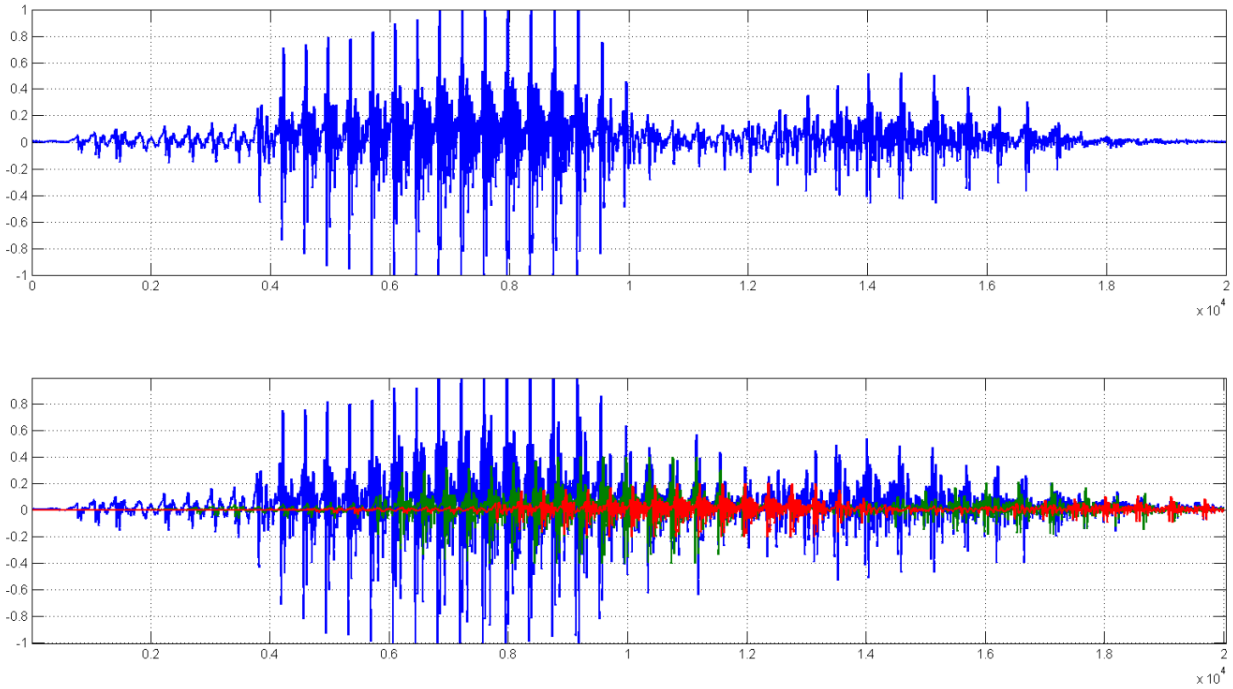

Figure 5.3 – a) Original speech signal not affected by reverberation; b) Speech signal affected by reverberation and reflected components

If the reverberation adds components to the sound signal that are difficult to remove, the sound source movement will make the problem exponential more difficult. This is because the parameters that describe the movement of the sound source become another set of unknown variables. Now, in the case of speech signal mixes the unknown information is:

- The number of speech sources and associated variables (gender, intensity, language, accent, speed, etc.)
- The exact time when each of the persons speak
- The location of each speech source within the room
- The room's geometry
- The trajectory and speed of each speech source

Adding noise to the recorded sound will also require more complexity from the analysis method.

In the following section we will attempt defining the speech signal mixes and the process of separating each source from a mathematical point of view. This is necessary to create a common ground for research. The paragraph will describe concepts like convolutive mixture, instantaneous mixing, delayed sources, etc. In the following paragraphs we will highlight some of the methods used in state of the art BSPS systems, for using multiple or single microphones.

5.2 PROBLEM DEFINITION FORMALISM

The simplest mixture of speech signals is defined by Scot C. Douglas and Malay Gupta in [Makino, 2007] as instantaneous mixing. Figure 5.4 shows a basic diagram that symbols BSPS of an **instantaneous signals mix**. $\{s_i(k)\}$ are a set of m unknown source signals. The task of B(k) system is to reconstruct as best as possible the s(k) set of signals. The mixing system – A – will mix the speech signals and produce an output signal. The output signal can be measured by a set of n sensors, producing $\{x_j(k)\}$ set of signals. The x(k) signals can be affected by noise – v(k) in our case. To simplify the problem we consider that each of the x(k) signals is affected by the exact same noise signals. The y(k) set of signals represents the reconstructed set of signals.

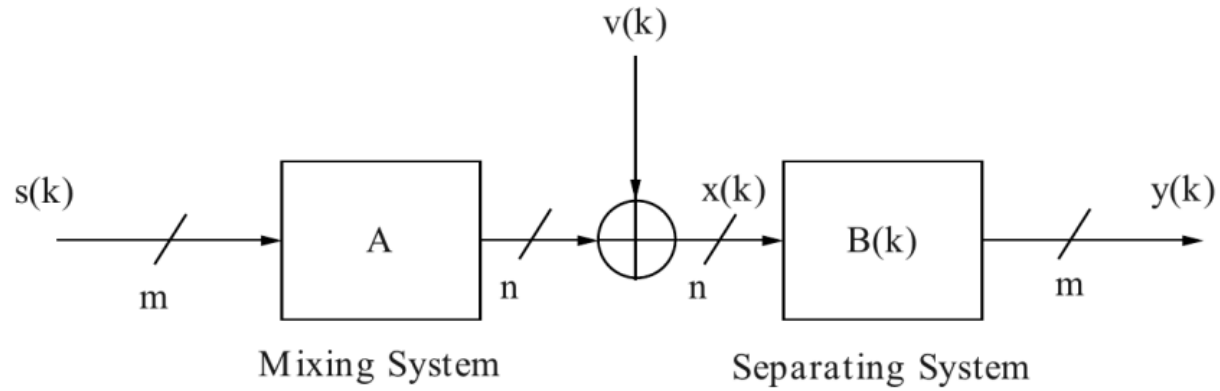

Figure 5.4 – Instantaneous mixing diagram

We can define A mixing system as a *(n x m)* matrix A invariant in time with $\{a_{ij}\}$ coefficients. In this case, we can define $x_j(k)$ signal as in (5.1).

$$x_j(k) = \sum_{i=1}^{m} a_{ji} s_i(k) + v_j(k) \tag{5.1}$$

The objective of a BSPS task considering instantaneous mixtures is to find a set of *(m x n)* coefficients for a B matrix, representing the de-mixing system in order for *y(k)* set of signals defined in (5.2) to approximate as good as possible the *s(k)* set of source signals.

$$y_i(k) = \sum_{j=1}^{n} b_{ij}(k) x_j(k) \tag{5.2}$$

However, instantaneous mixing is not usually encountered in practice mainly because this approach ignores the acoustic paths. In order to achieve reliable separation results in state of the art systems, models for separating convolutive mixtures must be implemented.

The concept of **convolutive mixture** takes into account the reverberation phenomenon described in the above section. For each sound source a number of **delayed sources** is generated due to reflections of sound of various objects in the room. In this case, figure 5.4 can be adapted and the time-invariant A mixing system will become a time-dispersive multichannel system.

$$x_j(k) = \sum_{l=-\infty}^{\infty} \sum_{i=1}^{m} a_{jil} s_i(k-l) + v_j(k) \tag{5.3}$$

Equation (5.3) describes the mixing process. Basically we consider an infinite number of reflections that make up the measured *x(k)* signal. Due to this added complexity, algorithms that target BSPS need to take into consideration the following:

- Spatial unmixing
- Temporal changes introduced by the mixing system

Algorithms that exploit the mentioned aspects – spatial and temporal signal characteristics – are called spatio-temporal BSPS algorithms. Finally, *y(k)* set of signals are defined by (5.4) and need to encompass an approximation as good as possible of the *s(k)* set of signals.

$$y_i(k) = \sum_{l=-\infty}^{\infty} \sum_{j=1}^{n} b_{ijl}(k) x_j(k-l) \tag{5.4}$$

The ambiguity that a speech source separation system needs to face is that the *m* number of sources is unknown. Many algorithms assume the correct value of *m* or try to determine it. This thesis also proposes an algorithm for determining the value of *m* in a speech mix. Another approximation that needs to be taken into consideration is limiting the number of reflections – delayed sources that participate in the mix. If we consider a high number of A matrices when dealing with convolutive mixtures, the problem's computational requirements increase greatly. In general, algorithms limit the number of delayed sources.

5.3 SPEECH SIGNAL PROPERTIES

Speech signals have a number of properties that contribute to simplifying the problem of separating them. Many algorithms rely on the following 3 assumptions that cover most practical cases:

- Speech signals are statistically independent, considering they are produced by different speakers at different locations in space, in the same acoustic environment;
- We can consider speech signals as quasi-stationary for reduced time durations (around 10-15 milliseconds) but for longer periods they are nonstationary.
- Each source has a unique structure in time over short time frames

[Makino, 2007] states that the 3 principles mentioned above can be used by a separation system but theoretically it is possible to have a system targeting BSPS and that uses only a single statement.

5.4 MICROPHONE ARRAY SPEECH SOURCE SEPARATION

The mathematical model illustrated by (5.4) shows that the signal *x(k)* resulting from the mixing process can be measured with a set of n sensors (microphones). When *m*, the number of speech sources is greater than *n* the problem is called **underdetermined**, meaning that the measurement setup does not use at least one microphone per source. When the number of microphones is greater than the number of sources *s(k)* the problem is **overdetermined**.

Current state of the art BSS systems tend to use microphone arrays to measure the signal mix, in order to improve accuracy. Some practical applications like biomedical signal processing, geophysical exploration allow and even more – require multiple microphones to be used. However, when the analysis device becomes smaller in physical dimension, the number of microphones that can be employed reduces. Current flagship smartphones use more than one microphone for tasks like recording, speech recognition, etc., but the concept of personal assistant (a wearable device that does various aiding tasks – including speech recognition) becomes more widespread researchers might need to invest in the BSPS done with a single microphone.

According to [Puigt, 2011] the methods aiming for BSPS using microphone arrays can approximately be grouped in 3 classes *when facing instantaneous mixtures*:

- Independent Component Analysis (ICA) – Sources are statistically independent, stationary and at most one of them is Gaussian;
- Sparse Component Analysis (SCA) – Sources are sparse, meaning that most of samples are null or close to zero. Regular speech tends to resemble to this case where speakers cannot speak continuously;
- Non–negative Matrix Factorization (NMF) – These methods handle the case where both sources and mixtures are positive, with possibly constraints over scarcity.

Other methods for BSS include:

- Principal component analysis
- Singular value decomposition
- Dependent component analysis
- Low-complexity coding and decoding
- Stationary subspace analysis
- Common spatial pattern.

Classifying the methods of BSS is often difficult because there are many parameters to take into account, like the type of mixtures they operate on (convolutive or instantaneous), the type of problem (overdetermined, underdetermined) the assumptions they make on the sources (sparse, stationary, statistical independent), the domain they operate on (frequency or time), etc. In the current sub-sections the focus will be on some of the most cited methods in reference publications: independent component analysis (ICA), FastICA (fast independent component analysis) and SCA.

5.4.1 Independent Component Analysis (ICA)

A BSS method called independent component analysis was developed by Aapo Hyvarinen, and first presented in [Hyvarinen, 1999]. In [Hyvarinen, 2000] the work was further more explained. These papers paved the road for various algorithms for BSS and one of them is FastICA. The main idea of ICA method is to determine the coefficients of the mixing system *A* so that the resulting *s(k)* set of signals respect some imposed constraints.

Consider that *x* is a set of *N* observations produced by mixing *N s* sources using *A* mixing system:

$$x = As \tag{5.5}$$

To simplify we consider the problem to be determined, meaning that the number of sensors recording the signal mix is equal to the number of sources. Since the goal is to reconstruct the speech sources, (5.5) can be rewritten in (5.6).

$$s = A^{-1}x = Wx \tag{5.6}$$

In (5.6) *s* and *W* are unknown variables. However, we know that s needs to respect a set of rules as described in section 4.3 and as they will be detailed in the following paragraphs. In computational language, the ICA algorithm can be reduced to the following steps: matrix W is given a set of initial values, using (5.6) the set of source signals is computed, the resulting signals are checked to discover if they respect the set of imposed constraints, if the constraints are respected then the solution is saved – if not the W matrix has its coefficients adjusted. Lets now look over the set of constraints that need to be respected by the source signals: as stated in [Hyvarinen, 2000] the variables must be nongaussian and independent.

The first thing needed is a metric to assess the degree of nongaussianity of a random variable, denoted *v* in our case. [Hyvarinen, 2000] proposes 2 metrics: kurtosis and negentropy. **Kurtosis** can be defined as in (5.7).

$$kurt(v) = E\{v^4\} - 3(E\{v^2\})^2 \tag{5.7}$$

If v variable has the variance equal to 1, (5.7) can transform into:

$$kurt(v) = E\{v^4\} - 3 \tag{5.8}$$

The kurtosis can be negative of positive. In the first case the *v* variable is said to be subgaussian while in the latter the *v* variable is supergaussian. The interest – in order to have a clean separation of sources – is to have the kurtosis of the random variable equal to 0. In this case, the search for a set of signal sources and mixing system coefficients ends when the kurtosis of the determined signals are null and the estimated signals respect the *independence* property. The independence property is simplified by (5.9) that is true only when x_1 and x_2 are independent random variables:

$$kurt(x_1 + x_2) = kurt(x_1) + kurt(x_2) \tag{5.9}$$

The advantage of using kurtosis is that it is simple from the computational point of view but it also has some disadvantages when used in practice. According to [Huber, 1985] analysis using kurtosis can be sensitive to outliers – meaning that its value may depend on a few samples towards the end on the distributions. Therefore, other metrics of nongaussianity can yield better results. One of them is **negentropy**.

Using negentropy to determine if a random variable is Gaussian relies on the fact the entropy of a give signal is higher as the degree of unpredictability of the signal grows which results in the fact that gaussian random variables have high entropies. The entropy of a given random variable is defined in (5.10).

$$H(V) = -\sum_i P(V = a_i) \log P(V = a_i) \tag{5.10}$$

For practical reasons we need a metric that is zero when the random variable is Gaussian so therefore we must use (5.11) as defined by [Cover, 1991], to introduce the differential entropy or a random variable v with f(v) density.

$$H(v) = \int f(v) \log f(v) dv \tag{5.11}$$

Further on (5.12) describes the J metric, called negentropy which is 0 when the random variable is Gaussian.

$$J(v) = H(v_{gauss}) - H(v) \tag{5.12}$$

In (5.12) v_{gauss} is a Gaussian random variable of the same covariance matrix as v. As [Hyvarinen, 2000] states, the advantage of using negentropy or differential entropy to determine the nongaussianity is justified by statistical theory. However, negentropy computation is generally difficult from the computational point of view; therefore, some methods of **approximating negentropy** are used. A classical method for approximation is presented in [Jones, 1987] and defined in (5.13). The formula uses kurtosis metric and higher order moments when the v variable is assumed to be of zero mean and unit variance. This approach is however limited because it inherits the lack of robustness brought by computation of kurtosis.

$$J(v) \approx \frac{1}{12} E\{v^3\}^2 + \frac{1}{48} kurt(v)^2 \tag{5.13}$$

[Hyvarinen, 1998] introduces another method for approximating the negentropy in (5.14) using a set of nonquadratic functions G_i. The pair of function expressed in (5.15) is claimed to have been proved useful by [Hyvarinen, 2000]. We can observe in (5.14) the addition of w term which is a Gaussian variable with variance equal to one and zero mean.

$$J(v) \approx \sum_{i=1}^{p} k_i [E\{G_i(v)\} - E\{G_i(w)\}]^2 \tag{5.14}$$

$$\begin{aligned} G_1(u) &= \frac{1}{a_1} \log \cosh a_1 u \\ G_2(u) &= -\exp(-u^2/2) \end{aligned} \tag{5.15}$$

The described approach of estimating negentropy gives a good compromise between metrics given by kurtosis and negentropy. Getting back to the main goal where we need to estimate the coefficients of the W matrix and s signals, the convergence point is reached when $J(v)$ negentropy is close to 0.

The steps presented above described by formulae (5.7) – (5.15) have been developed into faster methods for approximating W matrix. For example, in [Pham, 1992], the model is estimated by a maximum likelihood method, defined in (5.16).

$$L = \sum_{t=1}^{T} \sum_{i=1}^{n} \log f_i(w_i^T x(t)) + T \log |\det W| \tag{5.16}$$

In this section measures of nongaussianity were defined (kurtosis and negentropy) along with methods to approximate the negentropy. By minimizing these measures we can verify that the W and s value sets were determined correctly. This task implies successive adjustments over the coefficients of the W matrix. However, this chapter does not describe a strategy to minimize the number of adjustments. The following chapter describes FastICA algorithm. This is also presented in [Hyvarinen, 2000] but has been used widely ever since its release, with implementations existing in a wide spectrum of programming languages C, C++, Matlab, Octave, R as seen in [FastICA, 2000].

5.4.2 FastICA

The FastICA algorithm is used for determining ICA components in a fashion with reasonable computational requirements. However, it poses some constraints over the input signals. It requires that the signals are **centered** and **whitened**. Centering requires that the *x* signal is a zero mean random variable. This means that $m = E\{x\}$ is subtracted from *x*. A signal is whitened when its covariance matrix equals identity. Equations in (5.17) describe the whitening process. The advantage is that whitening reduces the number of parameters to be estimated because the *A* mixing matrix becomes orthogonal.

$$\begin{aligned} &E\{\tilde{x}\tilde{x}^T\} = I \\ &\tilde{x} = ED^{-1/2}E^T x \\ &\tilde{x} = ED^{-1/2}E^T As = \tilde{A}s \\ &\tilde{A} = ED^{-1/2}E^T A \end{aligned} \tag{5.17}$$

In (4.17) E is the orthogonal matrix of eigenvectors $E\{xx^T\}$ and *D* is the diagonal matrix of its eigenvalues, $D=diag(d_1, \ldots, d_n)$.

So, before FastICA computations start, *x* input signal needs to be whitened and also a whitened variant of *A* matrix will be computed, which needs to be transformed in post-processing in the required *A* mixing matrix. *Nevertheless, the algorithm can be improved by applying problem specific optimizations*.

FastICA uses the derivatives of the nonquadratic functions defined in (5.15) which are given by (5.18).

$$\begin{aligned} &g_1(u) = \tanh(a_1 u) \\ &g_2(u) = u\exp(-u^2/2) \end{aligned} \tag{5.18}$$

The steps of FastICA, considering that a single *w* vector is being searched, are described below:

1. w vector takes an initial set of random values
2. $w^+ = E\{xg(w^T x)\} - E\{g'(w^T x)\}w$
3. $w = w^+ / \| w^+ \|$
4. If not converged, reiterate step 2

In this case, convergence means that the dot product of old values and new values of *w* are almost equal to 1. Implementations of FastICA are – as stated before – widely available. The algorithm has also some interesting properties.

- Convergence is cubic (or at least quadratic). General ICA algorithms based on gradient descend had linear convergence.
- The algorithm does not use a step size meaning that it requires no extensive fine tuning making it easy to use.
- The *g* function can be any non linear function and can be tuned depending on the specific of the task.
- The algorithm can be easily parallelized and requires little memory space.

5.5 SPEECH SOURCE SEPARATION WITH KNOWN SPEAKER COUNT

The methods presented in chapter 4 can be integrated with a BSS algorithm. It can use the input from a single microphone in the setup or it can combine the results from simultaneous analysis of all microphone inputs. In case of the latter, the results can theoretically be improved by a simple voting mechanism. Therefore, after the speaker count estimation step, the BSS will already know the number of sources participating into the mix. An additional advantage is added if the separation problem is over-determined, meaning that the number of microphones is higher than the number of sources. In this case, multiple combinations of input signals can be made for source separation, with the possibility of selecting the best performing combination.

We propose the usage of pipelined speaker count detection, using multiple analysis windows of increasing size. As seen in [Andrei, 2014] and [Andrei, 2015] the detection performance increases with window size. This was also an important conclusion extracted from the analysis presented in section 4 of this work. A selector block must be implemented to decide which of the detectors will give the final output. The selector block must also implement a decision history because the bigger sized frame buffers also contain the previous smaller sized frame buffers.

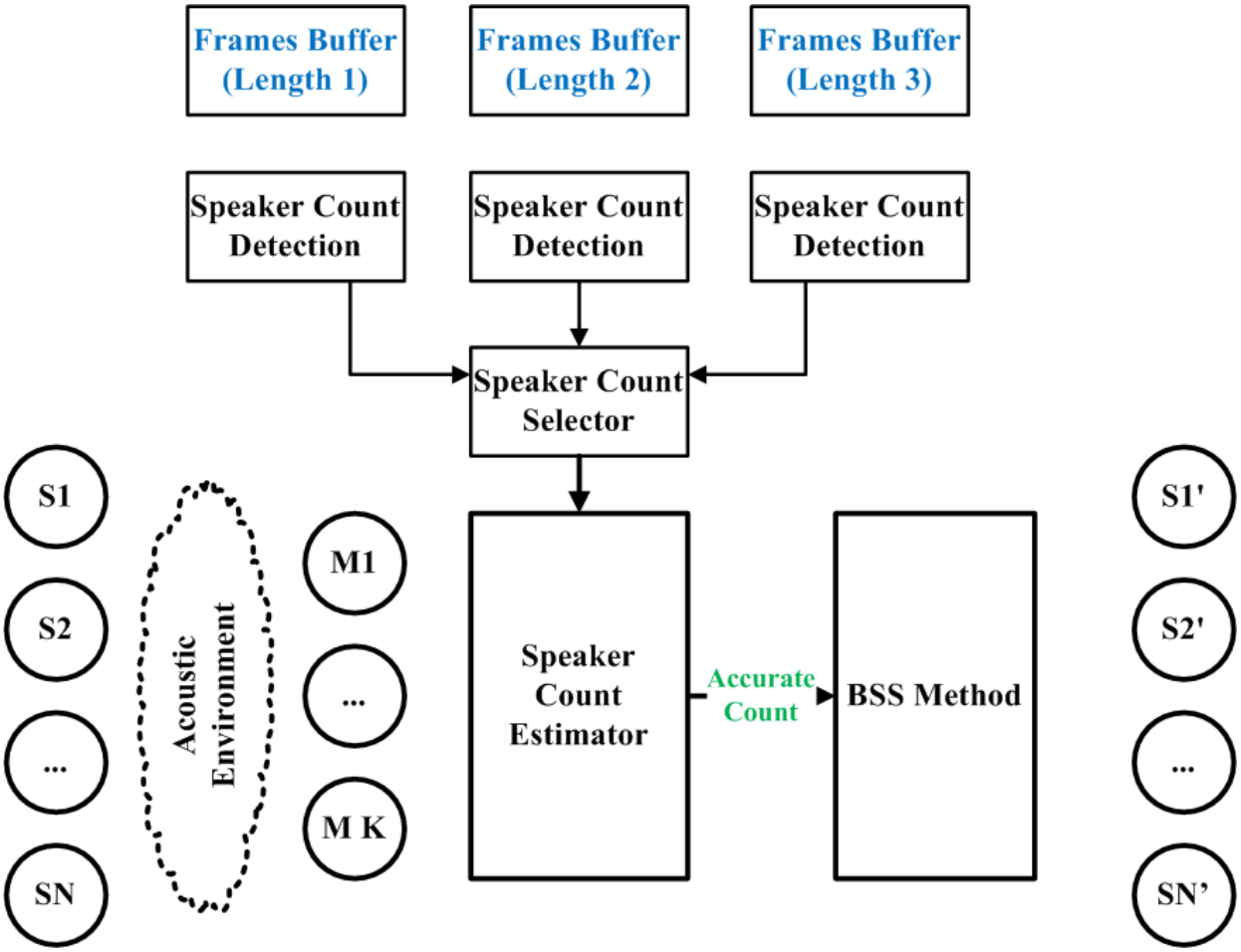

Figure 5.1 – Speaker count estimation integration with BSS methods

In the following experiments we used the speech mixtures used to evaluate the accuracy of the simultaneous speaker counting methods as input to the BSS methods. Therefore, the method presented in section 2 outputs the number of active sources to the BSS algorithm enabling the separation. We focused mainly on BSS methods derived from independent component analysis (ICA).

5.5.1 Algorithms used for separation

As stated before, there are a fair number of variations of ICA that try to perform the BSS task. In [Cichoki, 2003] the authors have gathered the main methods into a single Matlab package with a set of associated benchmarks, to evaluate each algorithm. The toolbox is called ICALab. We used three of the best performing algorithms in our experiments to separate the speech sources. We also added a non-ICA based method in order to compare separation quality. The used approaches are the following:

- **FastICA** – Used for determining ICA components in a fashion with reasonable computational requirements. However, it poses some constraints over the input signals. It requires that the signals are centered and whitened. Centering requires that the average of the signal is subtracted while whitening is done in order to reduce the number of parameters that need to be estimated by making the *A* mixing system an orthogonal matrix. The FastICA method is also described in [Cichoki, 2003].
- **FlexICA** – The method is called "flexible ICA" and was described in [Choi, 2000]. The approach is able to separate a mixture of heavy and light tailed sources using flexible non-linearity and learning un-mixing on the decorrelating manifold.
- **RADICAL ICA** – This algorithm presents a different approach for entropy estimation adding a relative insensitiveness to outliers. The method is described in [Miller, 2003] and is stated to yield superior performance to Fast ICA.
- **SOBI-RO** – The method is described in more detail in [Belouchrani, 1993] and exploits the possible time coherence of sources signals. This is not an ICA based approach but similarly to ICA methods it needs to know the number of sources before starting the separation process.

One practical disadvantage of ICA derived methods is that they are sensible to sound propagation related effects. For example, in a room that is not soundproofed the microphones will not only capture the speech sources but also a multiple number of their reflections caused by the room's walls. This effect is called reverberation and makes the BSS task much more difficult. Also, when using a microphone array, the sound will reach each of the microphones at different times, with larger delays at more distant microphones. This is why we added another algorithm, called STFICA, to our experimental setup. The approach is described in detail in [Douglas, 2007] but is stated to be more robust to reverberation and sound propagation effects.

5.5.2 Separation quality assessment

While the BSS task can be easily evaluated qualitatively judging the clearness of the separated sources, we need to define a metric for assessing the performance of a separation algorithm. One such method is called the *Amari distance* and was firstly introduced in [Amari, 1996]. The metric can be described by (5.19), where A_0 represents the ideal

instantaneous mixing system, A is the mixing system estimated by the BSS algorithms, p represents the number of components and finally $r_{ij} = (A_0 A^{-1})_{ij}$.

$$d(A_0, A) = \frac{1}{2p}\sum_{i=1}^{p}\left(\frac{\sum_{j=1}^{p}|r_{ij}|}{\max_j |r_{ij}|} - 1\right) + \frac{1}{2p}\sum_{i=1}^{p}\left(\frac{\sum_{j=1}^{p}|r_{ij}|}{\max_i |r_{ij}|} - 1\right) \tag{5.19}$$

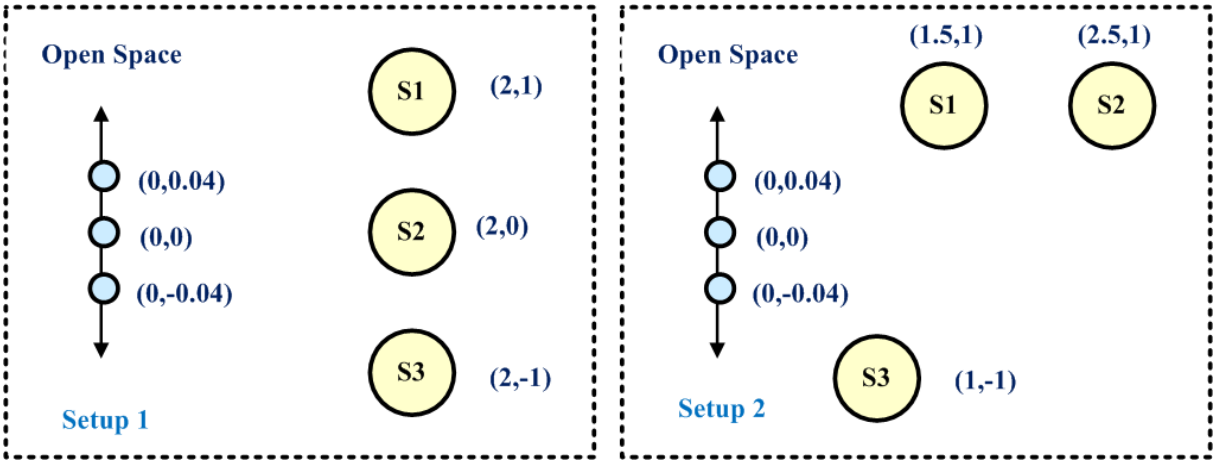

Figure 5.6 – BSS experiment speaker setup

The *Amari distance* operates on nonsingular matrices and yields values between 0 and 1, with 0 meaning perfect separation. This metric is a standardized approach to report performance of BSS algorithms. An important observation is that this metric is not symmetric and the order of operands needs to be the same for all the methods. We used as first operand in our evaluations, the ideal mixing matrix.

5.5.3 Experimental setup

Considering that all the BSS algorithms mentioned in the previous paragraphs are microphone-array approaches, we simulated a recording setup. Figure 5.6 illustrates our proposal. We have two different setups. The microphones were placed equally spaced across a line. We considered 3 microphones and used the maximum distance between them was 8 centimeters. This is a reasonable dimension for a smartphone that can use microphone arrays to improve speech separation quality.

We considered that the speakers are placed in open space so no reverberation effects will occur. However, we treat two distinct cased of combining the sources. The simplest case is where the speech sources are combined instantaneously, ignoring the reception delays due to sound propagation but taking into consideration the introduced attenuation. The challenging case that will yield higher errors is where we also take into account the propagation delays. For both cases we considered an atmospheric absorption factor of 2.73 dB/km corresponding to a 20^{o}C temperature, 50% humidity and an average speech frequency of 500 Hz. The results produced by the algorithms mentioned in section 3, can be analyzed in Table 5.1.

For the experiments we used 100 mixtures per each setup and for each algorithm. The mixtures contained mixed speech from one to five seconds. We did not see any accuracy variation with the length of the speech mixture – the one second mixtures were separated with the same accuracy as the five seconds recordings. It is also important to remember that before

combining the sources we eliminated the silence periods using the VAD provided by Matlab Voicebox.

By denoting the microphones $m_{i,j,k}$ and the speakers $s_{i,j,k}$ the signal acquisitioned by microphone j – x_j – for example is given by (5.20). The variable A represents the atmospheric sound atteniation while v_s is the sound speed under normal atmospheric conditions.

The *trim* operator selects only the first *N* samples from a signal. We use it to simulate the delay perceived by the microphone, caused by the time needed for the sound to reach the microphone. We create a vector with a number of zero elements resulting from the effective distance from the speaker to the microphone. If the speaker is more distant to the microphone, the number of zero samples will increase accordingly. (5.19) formula can be applied symetrically for x_i and x_k signals.

$$
\begin{aligned}
d_{jj} &= \sqrt{(s_{jx} - m_{jx})^2 + (s_{jy} - m_{jy})^2} \\
d_{ji} &= \sqrt{(s_{jx} - m_{ix})^2 + (s_{jy} - m_{iy})^2} \\
d_{jk} &= \sqrt{(s_{jx} - m_{kx})^2 + (s_{jy} - m_{ky})^2} \\
x_j &= Trim[zero(1:\ f_s \frac{d_{jj}}{v_s}); s_j](1 - d_{jj} \frac{1 - 10^{\frac{A}{20}}}{1000}) + \\
&Trim[s_i(1:\ f_s \frac{d_{ji}}{v_s}); s_i](1 - d_{ji} \frac{1 - 10^{\frac{A}{20}}}{1000}) + \\
&Trim[s_k(1:\ f_s \frac{d_{jk}}{v_s}); s_k](1 - d_{jk} \frac{1 - 10^{\frac{A}{20}}}{1000})
\end{aligned}
\tag{5.20}
$$

The equations in (5.20) describe the mixing process. The resulting $x=[x_i; x_j; x_k]$ array is given as input to the FastICA algorithm, estimating the components one by one like in projection pursuit mode and *pow3* nonlinearity function.

5.5.4 Separation results

Instantaneous mixing was successfully de-mixed by all approaches except STFICA. We speculate that this was due to the fact that STFICA is designed for reverberating environments and was adapted using recordings taken into a closed room. The open-space environment was not favorable for the algorithm. The highest separation performance was achieved by the Radical ICA algorithm, with almost 2 times better Amari distance than its follower – Fast ICA. The hierarchy we achieved confirms the results published in [22].

When the propagation delays were included we see that the error of separation has increased drastically. We would have expected STFICA to perform better but it has only shown comparable results with the other 4 algorithms but with lower score. The best separation power was achieved by the SOBI-RO implementation.

A few observations need to be discussed. First of all we can observe that the positioning of speakers in space did not impact greatly the separation accuracy. This is a good indicator that the algorithms can generalize giving them the possibility of being used in other environments.

The speaker counting methodology was particularly helpful in cases where the Fast ICA and STFICA detected only one or two speech sources. This was mainly due to the fact that the covariance matrix had null elements therefore the number of sources were underestimated.

Given the fact that the speaker counting correctly estimated 3 sources, this served as a feedback and the input signals were amplified to eliminate high errors caused by floating point precision. Only Fast ICA and STFICA showed this behavior in the "whitening" phase critical at the beginning of the processing step.

The charts in figure 5.7 show that the Amari distance does not show sensitivity – that can be mapped to a trend – to the number of seconds of the mixture. This means that if the speaker counting methodology can operate on frames with reduced duration, the separation of sources does not turn into a bottleneck.

Table 5.1 – Amari distances for the BSS experiments

BSS Method	Instantaneous Mixing		With propagation delays	
	Setup 1	Setup 2	Setup 1	Setup 2
Fast ICA	0.034	0.034	0.303	0.317
Flex ICA	0.083	0.040	0.301	0.296
SOBI-RO	0.104	0.097	0.290	0.279
Radical ICA	0.015	0.015	0.303	0.309
STFICA	0.460	0.439	0.350	0.365

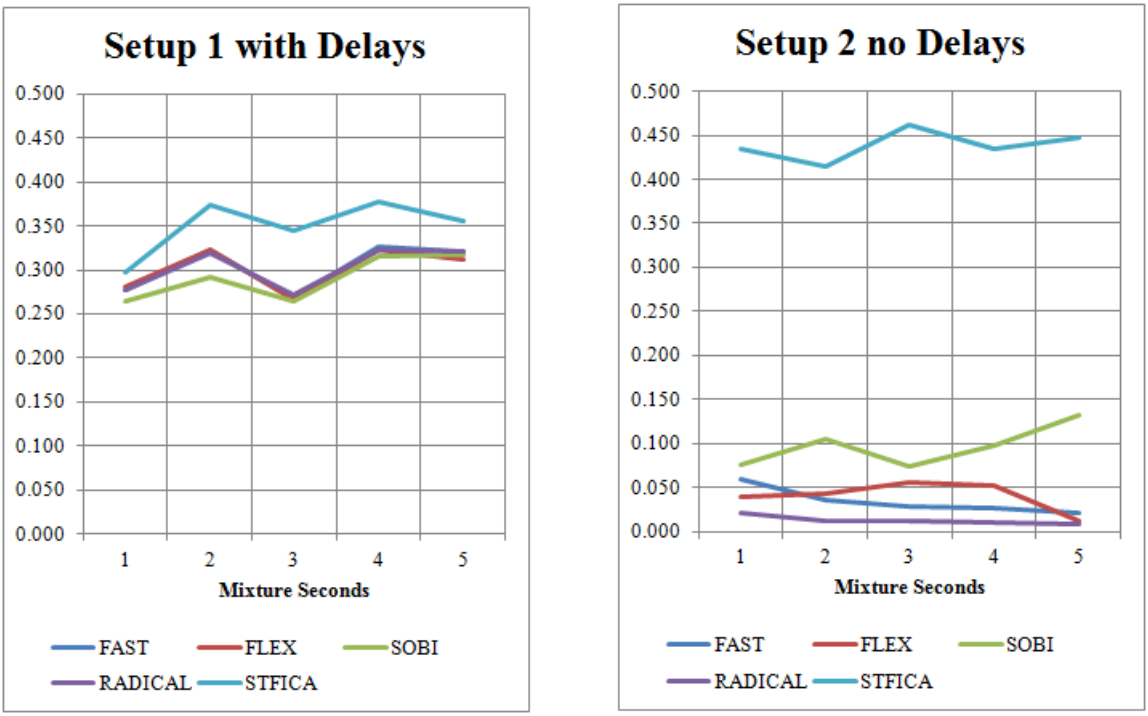

Figure 5.7 – Amari distance variation depending on mixture seconds

5.6 SUMMARY

This chapter analyzed how a method that counts the number of competing speakers can enable blind speech source separation algorithms. We observed that the method can show errors less than 15% even for 10 simultaneous sources when using a number of 5 single speaker references. This approach can be used to improve BSS methods. We selected 5 BSS algorithms, out of which 4 were based on ICA, and measured their separation performance using Amari distance. For this test we used recordings extracted from the same corpus used to analyze the speaker counting methods. The feedback from the speaker counting method was used to force the extraction of the detected number of sources from the mixtures.

Considering the proposed test conditions, measurements show that the sources can be separated almost perfectly when using instantaneous mixtures, while when taking into account sound propagation delays, each source can be perceived as separated but still the other two can be distinguished in the background. We also observed that on average Radical ICA yields best separation in the case of instantaneous mixtures which confirms previous work. STFICA approach showed the lowest separation accuracy. As a first future work we plan a better simulation of the recording environment using room impulse response methodologies – as presented in [Lehman, 2008] – and evaluate the separation performance of single microphone BSS approaches and also microphone-array methods that do not necessarily rely on independent component analysis.

6 COMPETING SPEAKER COUNTING BY HUMAN SUBJECTS

One of the main "features" that people have for following the correct speaker is binaural hearing. This enables us to localize the spatial origin of the sound and focus our attention towards it. Binaural hearing is widely described in technical research studies being referred as to binaural processing, beamforming or spatial filtering. For example, [Brown, 2005] presents a good synthesis over the methods used for sound localization and in [Mactech, 2010] we can see that some of the widely used high-end devices use these techniques to improve reliability of speech recognition.

Binaural hearing is an essential function, but we are able to distinguish between multiple speech sources even when we perceive them as coming from same direction. For example, during a radio talk show, at some point it happens that all the guests speak simultaneously. If we stay focused we can understand what each of the speakers is saying. This ability is called selective auditory attention (SAA) or selective hearing (SH). SAA is being studied from the medical point of view and the main goal of the research is to document occurring neurological events. In [Coch, 2005] we can read a study completed with the help of 44 child and adult volunteers that tries to study selective attention using nonlinguistic and linguistic probes. In [Gomez, 2011] the results of a research project that aimed to identify the causes for SAA disorders are presented. Even if the topic is of considerable importance, we could not find data that gives details about the actual performance of human SAA.

The current chapter describes the results of the SAA stress experiments that were produced by a group of native Romanian speakers. We will focus on determining the maximum number of simultaneous speakers that a person can follow. From a neuroscience perspective, knowing this information could be a first step in developing a functional model for the SAA. Another important result of the study is that each person described how the active speakers were identified and we were able to generate an interesting statistic.

This is a complementary research done in order to compare the human selective hearing capabilities to the artificial methods presented in Chapter 4. The mentioned section described the development of a method to automatically count the number of competing speakers in a recording and the current chapter will highlight how human volunteers respond to the same challenge.

6.1 SPEECH SOURCES

This experiment was realized in two sessions with the first one being described in [Andrei, 2014] and the latter described in [Andrei, 2015]. We are going to study these experiments in parallel in order to highlight if the findings from the first session match the findings from the second session.

6.1.1 Session one of the SAA experiment

In the first session the focus was to determine how accurately a human listener can count the number of competing speakers in a recording. In order to answer this question we did not focus on the fact that if a person actually knows the voice of the speakers, this might help count more accurately the number of speakers participating in the mix. Due to this reason we selected 10 random speeches from the Internet, with various topics being discussed. Half of the recordings are produced by known speakers while the other half is produced by lesser known persons. We mixed the captured speeches as described in Chapter 4. Table 6.1 describes the used speeches and speakers in more detail.

Table 6.1 – Speakers and speech topics for first SAA experiment session

Speaker & Source (Romanian Language)	Length	Short Description
Amza Pelea – "O afacere olteneasca"	6 Minutes 26 Seconds	The recording is a monologue from a popular Romanian movie called "O afacere olteneasca". What is specific to this recording is the fast speech pace the speaker employs to captivate the audience. The speech tales about the speaker trying to make a deal by buying and selling eggs but finds out that his idea is very bad because he actually loses money.
Andrei Pleșu – Discourse about success	4 Minutes 51 Seconds	The speaker has a very strong voice and talks about what success means to him and tries to define a set of metrics to evaluate personal success. The speech was held at a corporate event about leadership and success.
Dan Puric – Discourse about feelings	4 Minutes 1 Second	A speech held on the stage of National Theatre in Bucharest about love and understanding between people. What is specific to this speech is the intense passion transmitted by the speaker.
Emil Hurezeanu – Education conference	2 Minutes 47 Seconds	A passionate speech about the status quo of the Romanian educational system. The speaker has a very calm but strong voice.
Gabriel Liiceanu – Discourse about goal setting	5 Minutes 32 Seconds	The speaker has a very strong voice and the speech was recorded at the same conference as speakers 2 and 6. Even though the topic of the speech is similar as the one discussed by speakers 2 and 6, his speech is original and does not use stereotypes.

Florin Talpeș – Discourse about business strategies	5 Minutes 20 Seconds	Speaker has a very calm voice but not very strong. However, he articulates the words very well. The topic of his speech is related to effective leadership and generally uses corporate like language. We amplified his voice to make it more detectable.
Traian Băsescu – Political discourse	4 Minutes 27 Seconds	Speaker is the President of Romania and holds a speech about his suspension. He has a very strong voice and uses politics words.
Valeriu Nicolae – Discourse about personal history	4 Minutes 34 Seconds	The speaker has a voice similar to speaker 1 and basically talks about his life. He uses very general vocabulary. Since the speech is held in front of an audience, occasionally we hear reactions of the audience.
Badea Rozana – Laughter importance discourse	4 Minutes 26 Seconds	This female speaker has a strong voice and discusses about the importance of laugh for small children. She uses general vocabulary and occasionally words in the semantic field of child caring. The speech pace is rather fast.
Oana Marinescu – Discourse about trust and responsability	4 Minutes 41 Seconds	Also, a strong feminine voice as speaker 9 but the speech topic is totally different. She discusses in corporate terms about responsibility and trust. The speech pace is normal.

Out of the selected speakers, the first 7 are publically known people, known for their TV appearances. Of course the amount of notoriety is not identical for all speakers (i.e., the Romanian president is probably the most known person from this selection).

Table 6.2 reflects the mixing process for the first set of recordings. We also highlight the motivation for each selection. We want to highlight that the women voices were only introduced for speaker counts higher than 8, in order not to bias the detection for lower speaker counts.

Table 6.2 – Speech mixtures for first SAA experiment session

Nr. of Speakers	Speakers Last Names & IDs	Motivation
2	2, 5 – Plesu, Liiceanu	Up until 7 simultaneous speakers we selected the candidates with similar voices. Also, until this point we have not introduced the female voices or the voice of speaker 1 which was very fast paced and could have introduced the risk of keeping the focus of the listener "locked" to his tale.
3	2, 3, 5 – Plesu, Puric, Liiceanu	
4	2, 3, 5, 8 – Plesu, Puric, Liiceanu, Nicolae,	
5	2, 3, 5, 8, 7 – Plesu, Puric, Liiceanu, Nicolae, Basescu	
6	2, 3, 5, 6, 7, 8 – Plesu, Puric, Liiceanu, Talpes, Basescu, Nicolae	
7	2, 3, 4, 5, 6, 7, 8 – Plesu, Puric, Liiceanu, Talpes, Basescu, Nicolae, Hurezeanu	
8	2, 3, 4, 5, 6, 7, 8 – Plesu, Puric, Hurezeanu, Liiceanu, Talpes, Basescu, Nicolae, Marinescu	At this point we introduced the voice of the female speaker that should be heard very clearly.
9	1, 2, 3, 4, 5, 6, 7, 8 – Pelea, Plesu, Puric, Hurezeanu, Liiceanu, Talpes, Basescu, Nicolae, Marinescu	At this point we added the fast paced voice of speaker 1 and will ask the subjects if they are focusing on only one speaker.

10	All speakers	At this point we added the voice of the second female speaker.

6.1.2 Session two of the SAA experiment

After analyzing the results obtained during the first session of the experiment, we realized that knowing the voice of the speakers in the mixture can help the affect detection. We will discuss in future sections about what form this impact take can. Nevertheless, we were interested to quantify this impact and therefore we produced another set of mixtures. The recordings were created in a soundproof room and involved non public persons. For the second session table 6.3 describes the topics discussed by each of the speakers involved in the second set of mixtures.

Table 6.3 – Speech topics for second SAA session

Speaker ID	Topic	Speaker ID	Topic
1	Speech about Theory of Relativity and Maxwell equations.	9	Study about career success for women
2	Speech about history of Islamic culture.	10	Educational material about biotechnology
3	Speech about the experiments of the particle accelerator present at CERN.	11	Educational material about history of soccer
4	Speech about social processes and social engineering	12	Educational material about Romanian history
5	News regarding European Union politics and international context	13	Educational material about Romanian geography
6	Speech about corporate management	14	Educational material about organic chemistry
7	Speech about virtual machines	15	Educational material about middle east languages
8	Educational material about history of mathematics		

Since we are not using in the second experiment publicly known persons, highlighting which of the speakers participated in the mixtures brings no real value to this study. It is important to state that the recordings presented in 6.3, were used for all experiments involving the development of the automated speaker counting method. We chose these recordings instead of the ones used in the first session because they were controlled much more precise, they were recorded with the exact same equipment in the same soundproof room.

6.2 VOLUNTEER SELECTION

For the **first experiment** we used a group of 31 male and female listeners aged between 21 and 37. According to [Werner, 2009], school aged children can be auditory selective but inflexible. The development of selective hearing continues during adolescence so at the age of 21, persons can present a well developed SAA. We selected volunteers that were within top 15% of students considering Romanian grading system and demonstrated good concentration skills. In addition, the participants to this study were motivated by making them compete on who has the better accuracy in detecting the right number of active speakers from each recording with the winner being rewarded.

A number of 38 persons participated to the **second perception study**. We divided the volunteers into 2 groups: A and B. Participants from Group A and B have never met before. All members from Group A know each other because they are co-workers. Group A had 19 volunteers: 10 of them generated the test speech recordings while the other 9 had listeners' roles. Group B gathered 19 persons that were only listeners, so the total number of listeners was 28. For this experiment session the listeners are aged between 22 and 32. Figure 6.1 highlights the separation of the speaker groups in the second experiment.

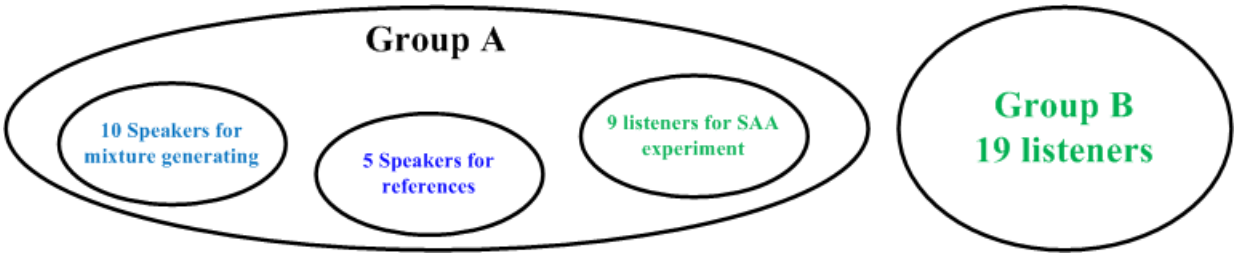

Figure 6.1 – Volunteer groups for seconds SAA experiment

6.3 EXPERIMENT METHODOLOGY

In **session one** of the experiment, the speakers had to listen to a recording without knowing the number of competing speakers and then when they were decided they would state the number of perceived speakers and additional observation. We also measured the thinking time for this session. The number of speakers in the mix was selected randomly and all speakers went through the same succession of speaker counts.

Session two of the SAA experiment was much more elaborate. We used customized software (*that was built specifically for this study*) and the speaker count succession along wing response time measurement was controlled much stricter. Figure 6.2 displays the graphical user interface of this software.

The software runs through the entire test but the listener also had the possibility of saving the current progress in case of fatigue effects. All the listeners were encouraged to give their best interest by using a scoring mechanism. The listeners also could give additional answers in support of the detected speaker count like the topics of discussion, known voices or additional observations. Let us discuss the steps of the perception experiment realized in the second session:

1. A set of pre-established listening durations were used: 5, 10, 20, 40 and 80 seconds.

2. We established a number of maximum 10 competing speakers.
3. For each competing speaker count and for each different listening duration we created a different mixture using the recordings provided by the 10 speakers from Group A, highlighted in figure 6.1. We obtained a number of 50 mixtures.
4. The 50 mixtures were considered different detection tasks for the listeners.
5. The software randomly selected a task from the set of 50 until all tasks were responded.
6. For each task, the listener had to input the perceived speaker count, and optional he was able to provide additional observations relevant to the experiment/

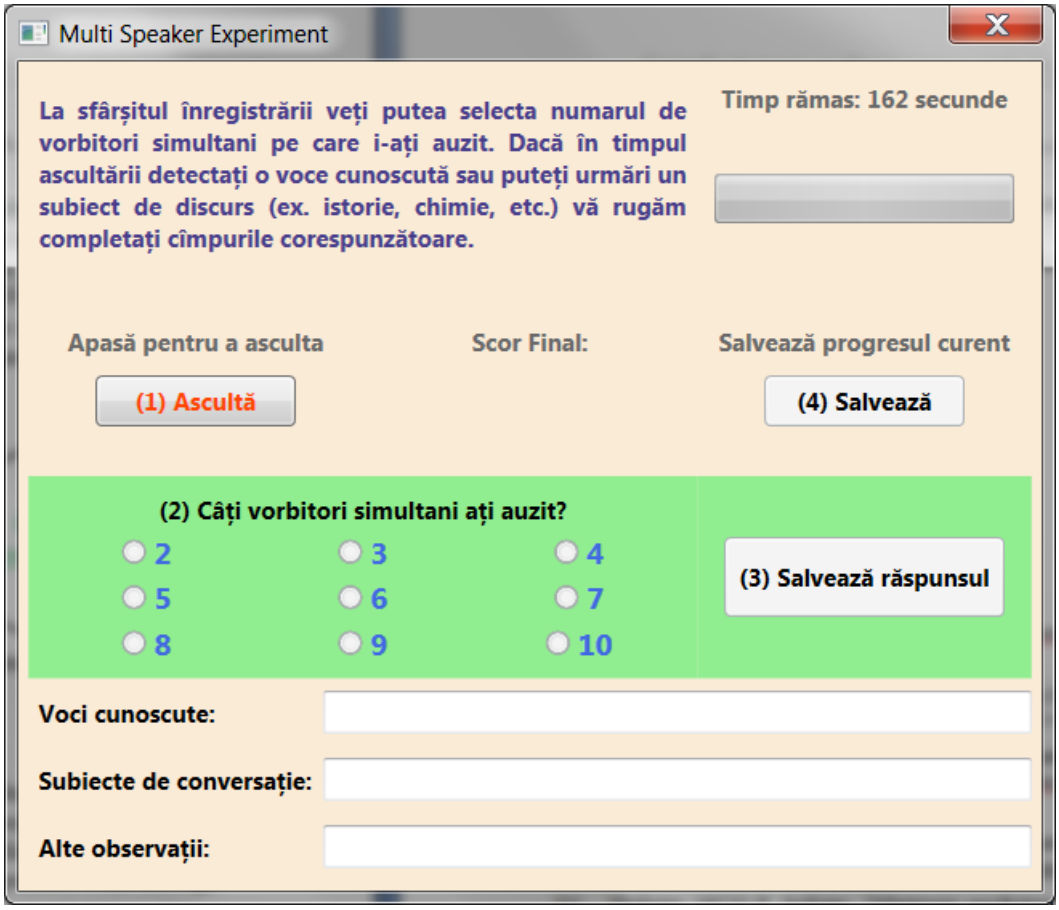

Figure 6.2 – Custom software for SAA experiment used in session two

We also mention that the log file generated by the application was saved in two different formats. The first format was a plain text file while the other one was an encrypted binary file containing the same information. By doing so we were able to check that the outputted results in the plain text file were not tampered with. Fortunately, none of the volunteers attempted this.

We encourage the use of software methods for these types of experiments as the amount of reproducibility of the setup for each participant is excellent, and the likeliness of outliers' occurrence is greatly reduced.

6.4 RESULTS OBTAINED IN THE FIRST SESSION

Perhaps the most interesting result is to see how many competing speakers can be detected by an adult person. In figure 6.3 each bar quantifies the percentage of listeners that counted incorrectly the number of active speakers noted on the x-axis. We can see that only 4% of the listeners gave wrong answers when presented a mix of 2 speeches but when the mix grew to 4 competing speeches, 54% were wrong.

When there were 5 competing speakers, less than a third of volunteers gave the correct answer. We can also see that the error grows somewhat logarithmically towards 100% showing that higher counts of simultaneously active speakers can easily confuse a human listener.

Each of the listeners was asked a set of 9 questions and we counted 87 correct answers from a total of 279. This means that the volunteers obtained a correct detection ratio for the number of competing speakers in a speech mix of 31%.

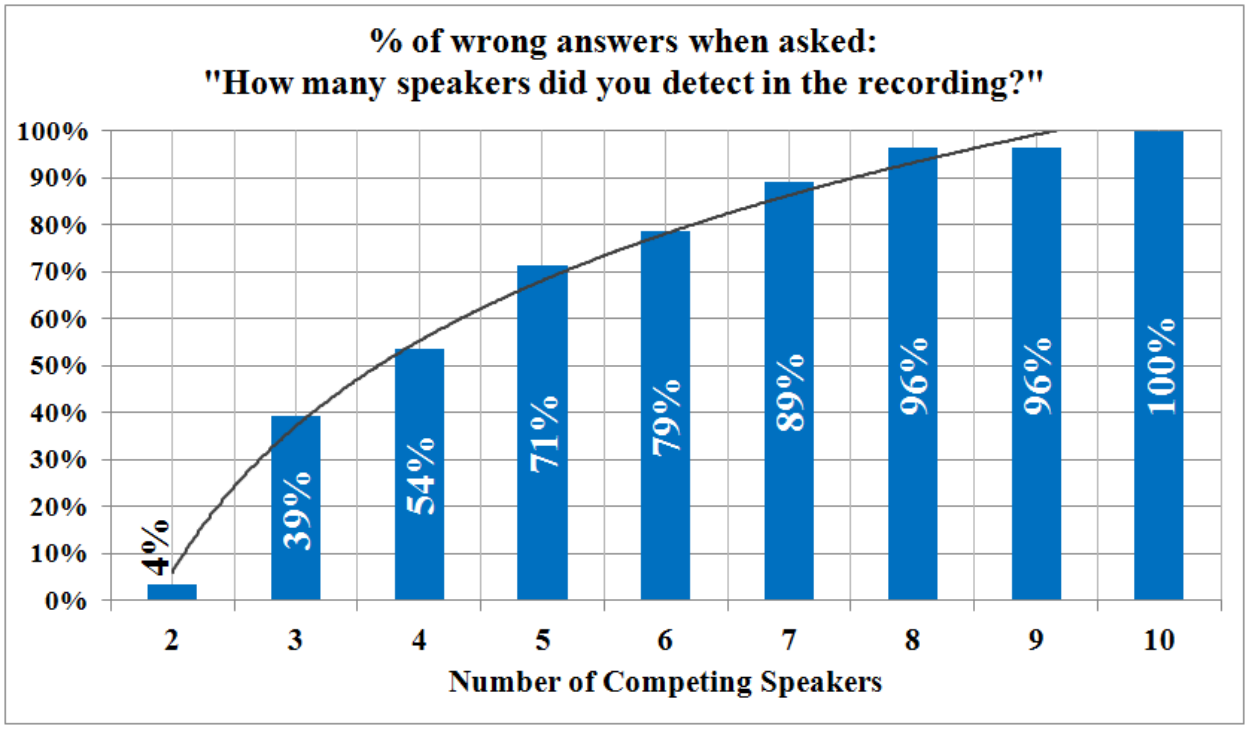

Figure 6.3 – Correct speaker detection for first SAA session

After the interview we asked each listener to describe how he detected the speakers in the recordings. The answers are aggregated in table 6.4.

The results are very interesting as they give clues on how SAA works. It is important though, to admit that the results have a moderate confidence level as each volunteer is unique can express in his own way the listening experience.

From table 6.4 we can see that the voice characteristic of a speaker plays a key role in SAA. In the same category we can include his gender. This is one reason for which we expect that the results presented in the second sessions will be more relevant.

A surprisingly low percentage of volunteers reported that they could follow speech – only 53%. Most of responders claimed that they just acknowledged pieces of speeches but they could not follow coherently a certain speaker. Data also show that the speech pace and

the silence periods were used as a factor in identifying new speakers. For example, some listeners reported that they heard a man talking rapidly and part of them said that they could follow that man because it kept their attention focused. The fact that 17% of listeners claimed to hear other languages than Romanian reflects the difficulty of the task, especially if the speaker count is higher than 5.

The next step is to analyze if the results produced by the second experiment are also confirming some of the findings presented in the first version of the study.

Table 6.4 – Listeners observations in session one of the SAA experiment

Observation / Method of detecting speakers	% of listeners used it
Learned voices from previous recordings	78%
Recognized a known voice	67%
Was able to follow transmitted information	53%
Recognized different genders	46%
Just guessed where speaker count was high	39%
Detected different speech paces	32%
Reported hearing other languages	17%
Used silence periods to identify new speakers	14%
Reported words that are repeating	10%

6.5 RESULTS OBTAINED IN THE SECOND SESSION

The second experiment was designed to answer some of the questions raised for the initial experiment. As stated in sections 6.2 and 6.3, we strictly controlled the listening duration and the quality of the speech samples to produce more accurate results.

Firstly, in figure 6.4 we can observe that the accuracy of the speaker count detection drops as the number of competing speakers grows. This is also an effect of the automated methods discussed in chapter 4 of this thesis. We can see that this effect becomes more pronounced for speaker counts higher than 4, which appears to confirm the results presented in the first experiment.

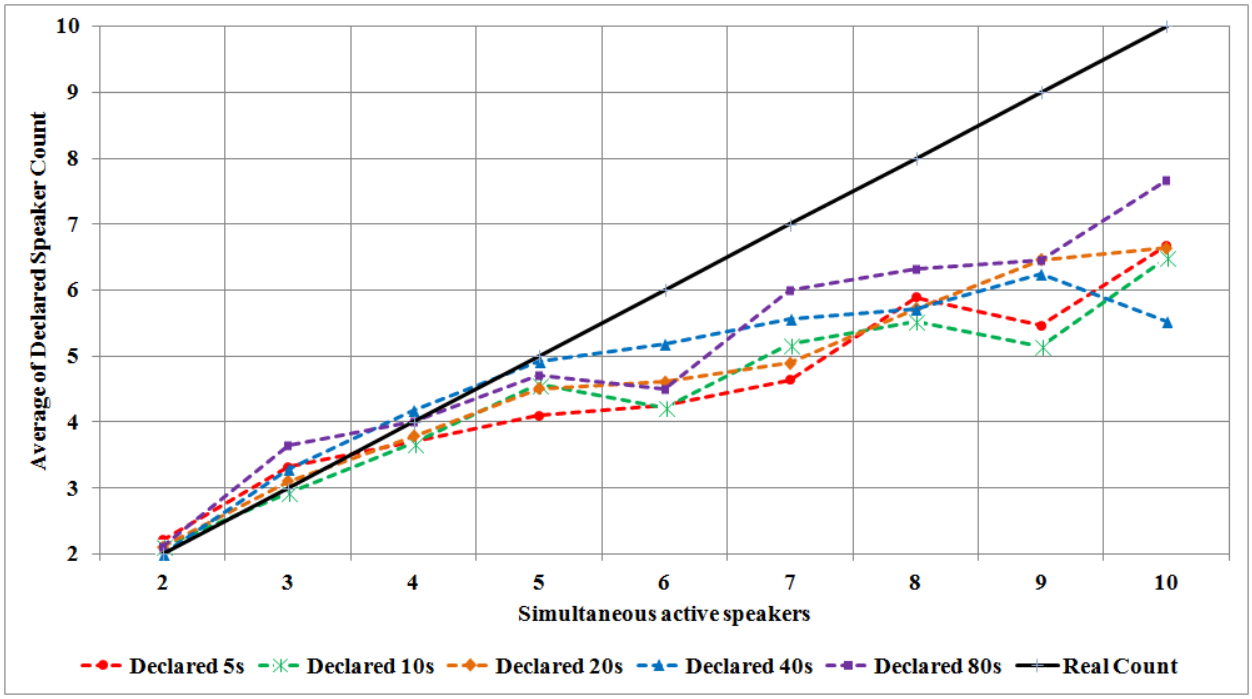

Figure 6.4 – Decrease of classification accuracy by human listeners with speaker count increase

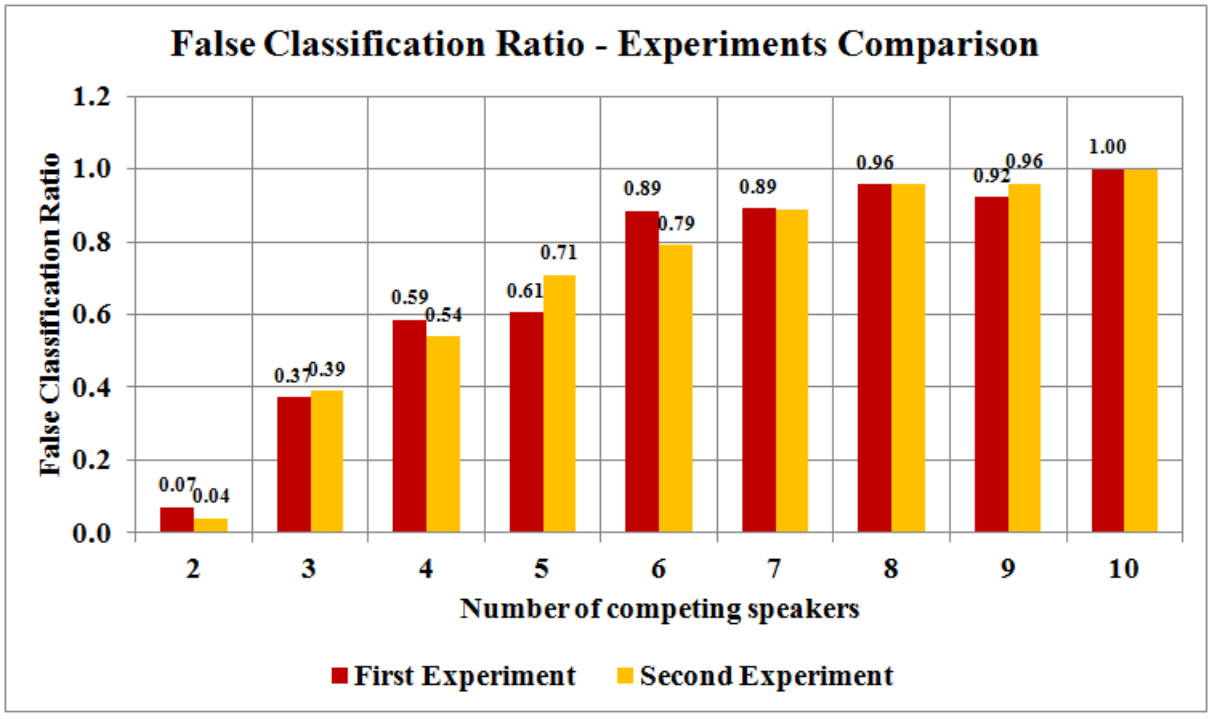

Figure 6.5 – FCR comparison between sessions of SAA experiment

Figure 6.5 highlights that the perception results produced by the two sessions of SAA experiment are extremely close. This can only strengthen the confidence in the findings. We can see that starting from 4 simultaneous speakers there is more than 60% chances that a human listener would guess the speaker count wrong.

Table 6.5 brings very interesting data. We see an obvious drop in False Classification Ratio (*previously defined in (4.16)*) if the listening time goes from 5 to 40 seconds. However, we can see that for all mixtures the FCR for 80 seconds is higher than the one for 40 seconds. We can only assume that this is an effect of fatigue and lack of concentration shown by the

listeners. So, time does influence the detection accuracy but not in all cases. It appears that listening to recordings for 40 seconds produces the most accurate results.

Table 6.5 – Influence of listening time over detection accuracy

FCR	2	3	4	5	6	7	8	9	10
5 seconds	0.14	0.39	0.64	0.64	0.96	0.93	0.89	0.93	1
10 seconds	0.11	0.29	0.68	0.50	0.86	0.96	0.96	1.00	1
20 seconds	0.07	0.46	0.64	0.71	0.89	0.96	1.00	0.82	1
40 seconds	0.00	0.21	0.43	0.43	0.79	0.82	1.00	1.00	1
80 seconds	0.04	0.50	0.54	0.75	0.93	0.79	0.93	0.86	1

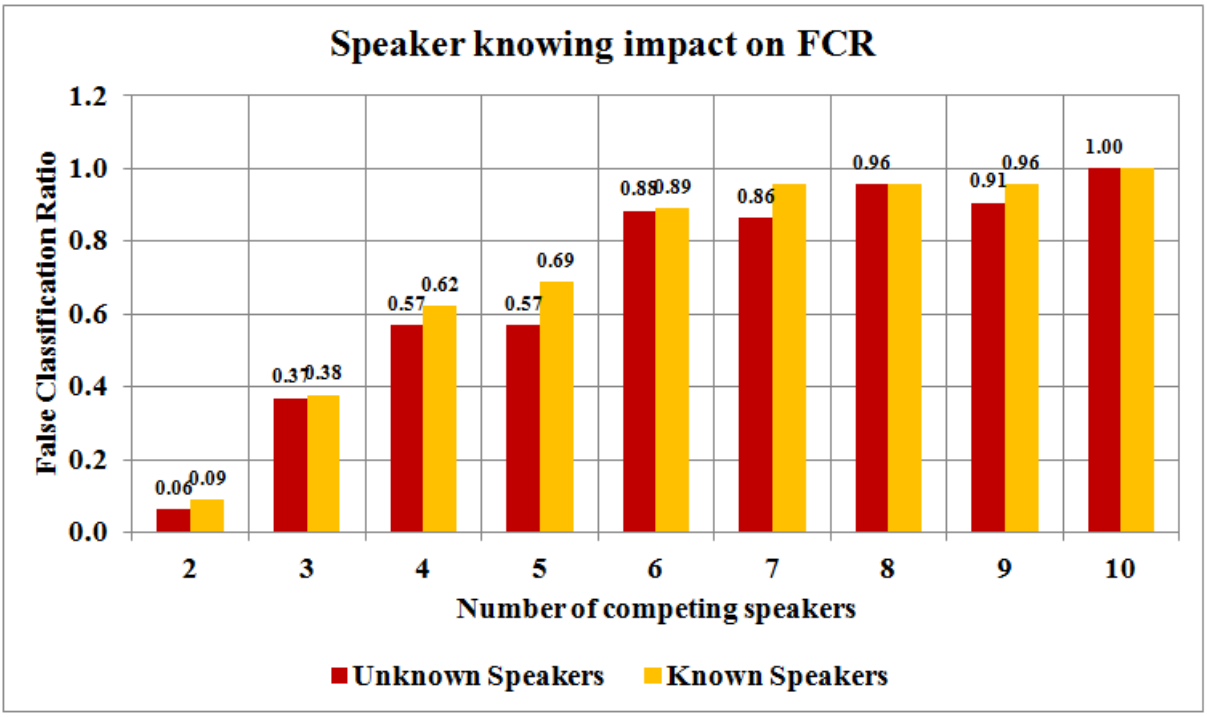

Figure 6.6 – Impact of knowing speakers over classification accuracy

In figure 6.6 we plotted the average FCR for each speaker count, considering the 5 variable duration recordings. The most interesting finding is that even if the mixes are created using familiar voices that does not improve the FCR. On the contrary we measured an average increase in detection error of 13%. Except for 3 simultaneous speakers, in all cases the detection accuracy was lower when familiar voices were used in the recordings.

The justification of this fact lies in the comments given by group A's listeners. Their main majority stated that when they were able to recognize a known voice, they continued following the detected speaker, "filtering" out all the other speakers. This can be viewed as a normal effect of human selective auditory attention. Group B listeners were not able to recognize voices therefore they were more focused on detecting as many speakers as possible. So, the lower FCR achieved by group A is explained as follows: even if group A has the advantage of knowing the voices in the recording, the listeners have the disadvantage of being distracted by recognized voices, not focusing on detecting other speakers, as the members of group B do.

6.6 SUMMARY

In this chapter we presented a perception experiment realized with a help of more than 30 volunteers. The first session of the experiment highlighted that for mixtures created by more than 4 simultaneous speakers, human listeners show great difficulty of classification. The same experiment revealed that the voices involved in the mixtures might play an important role in classification accuracy.

To strengthen the confidence of the findings, a second experiment was realized but with improved control over methodology. We used custom created software that accurately controlled the response time for each listener statement. We also used a set of improved recordings in the sense that they were collected with a high quality recording device in a soundproof room. For the second session we also divided the listeners into two groups to test if the correct detection of speaker count is dependent to knowing the voices involved in the mixture.

The second session of the experiment correlated with the results obtained in the first session. In addition, it demonstrated that listening time can influence positively the classification accuracy but only until fatigue effects occur. The best classification performance is at 40 seconds. For 80 seconds of listening time we see a classification error increase. The same study demonstrated that knowing the speakers involved in the mixture does not necessary increase detection accuracy. This is mainly because when a known speaker is detected it “shadows” the other detected speakers and captures the entire attention of the listener.

7 CONCLUSIONS

7.1 GENERAL CONCLUSIONS

Creating a large vocabulary continuous speech recognition system is challenged on multiple planes that can be considered standalone research directions. These topics mentioned in section 1.2 and addressed in more detail in chapter 2 include: phone recognition, accurate language representation, auditory scene analysis and improving robustness to accent, emotions, jargon and others. The objective of this thesis was to contribute to computational auditory scene analysis and more specific to competing speakers' environments.

We set the goal of developing a method that would accurately compute the number of competing speakers in each captured timeframe. This objective can be linked to voice activity detection methodologies, widely discussed in chapter 3 or it can be considered a new challenge by itself. In chapter 4 we highlighted some existing work where authors try to determine the number of competing speakers in a signal frame, using various methods ranging from statistical signal processing to audio-video analysis of the auditory scene. We considered that the video analysis of the auditory scene would raise numerous challenges in adopting proposed methodologies and, therefore, we focused in creating a method that would rely solely on signal processing.

The main philosophy behind the competing speaker counting methods is to quantify the amount of information carried by a speech mixture. We firmly state that as the number of competing speakers in a recording grows so does the information being carried out. A linked statement that leads to the development of the methods is that a certain signal frame that was produced by a number of competing speakers is likely to be more similar to a set of frames produced by the same number of speakers.

By trying to find solutions to the above questions we designed the proposed methods. As a first step we selected a set of simple metrics like signal power, active level, zero crossings rate that were combined with more complex metrics to produce an estimation of the number of competing speakers. Such metrics are derived for example from the estimated power spectrum and from similarity computation to a set of references. The metrics were combined and analyzed statistically or multiple samples were created to train and test a neural network.

We achieved a 17.2% False Classification Ratio when analyzing a set of 5000 frames of 5 seconds in length using a statistical tagging method presented in section 4.5 while if we consider a 20 seconds long frame, the error drops to only 1.2% . The neural network approach

reduced the FCR to 12% for the 5 second frame revealing that a learning based method can learn to select the most relevant features that can be used to tag a signal frame with a certain number of competing speakers. For these experiments we achieved the error rates considering that 50% of the references are also involved in producing the mixtures. This is actually a likely real case scenario as the participants to a conversation do not usually vary extremely.

However, in section 4.6, we tried to develop a methodology that would be more robust to references produced by speakers not participating in the mix. The previous discussed methods revealed a 70% error rate when using unknown frames while the time-frequency alignment method that uses Dynamic Time Warping can reduce the FCR to only 47%, where both numbers were computed on 5 second frames. However, the proposed method can show up to 3% FCR when the analysis frame is 25 seconds. This makes it possible for equipping a continuous speech recognition system with auto tuning capabilities making it possible to capture the single speaker speeches and combine them to produce the more accurate results mentioned for the previous methods.

The fact that we developed a method for determining the speaker counts in a timeframe, can aid speech separation algorithms. In chapter 5 we studied a set of algorithms based on microphone array separation of sources. The algorithms were extensions of independent component analysis approaches and are known to operate only when the number of sources participating to the mixture is fully known. In the same chapter we showed how various ICA based algorithms can successfully separate 3 single speakers from an instantaneous mixture with close to 0 Amari distance, while in the case where propagation delays are being encountered, the separation is successful but shows increased Amari distance.

If an artificial method can demonstrate 12% false classification ratio when analyzing 5 seconds frames, we wanted to investigate how difficult is this task for a human listener. Therefore, the 6th chapter is fully dedicated to a set of experiments designed to identify how accurate a group a volunteers can detect the numbers of competing speakers. We surprisingly discovered that human listeners are not good at correctly identifying more than 4 competing speakers. We observed that listening time can marginally reduce the identification error but only until fatigue installs. Also, surprisingly, when detecting known voices the accuracy drops because a known voice tends to capture the listener's focus, shadowing the other speakers participating to the mix.

7.2 PERSONAL CONTRIBUTIONS

The personal contributions brought by the thesis's author can be summarized as follows:

1. The thesis demonstrates that in a single channel recording, there is a direct relation between the number of competing speakers and the absolute value of various extracted features like signal power, zero crossings rate, entropy and distances to various single speaker references. In other words it is generally expected that a selected feature would evolve following a monotonic trend as the number of competing speakers grows or decreases. To our knowledge this finding was not mentioned or demonstrated in related work. The finding can be exploited to design methods for automatic participants counting in multi-speaker recordings.

2. A notable contribution is the analysis to determine how suitable are a set of features like signal power, zero crossings rate, entropy, spectral derived metrics and others for associating a number of competing speakers to an input frame. This was presented in section 4 of the thesis.

3. The algorithms for statistically classifying features of input frames into speaker count groups are also a contribution. Section 4 described a threshold based methodology for classification. This was followed by a more elaborated neural network based methodology that demonstrated superior accuracy. The analysis was enabled by a complete set of Matlab scripts and automated analysis tools developed by the author.

4. Also, in section 4 another approach for counting competing speakers in a timeframe was proposed. The methodology used reference signals (single speakers) and relied on the fact that as the number of speaker grows so does the distance to the references. The methodology uses dynamic time warping to classify input frames. The analysis was also enabled by Matlab scripts and optimized native DTW implementations which were developed by the author.

5. The contributions described in bullets 3 and 4 enable the functioning of independent component analysis based methods for blind speech source separation. Chapter 5 presents a set of ICA derived algorithms that were used to separate a set of instantaneous mixtures between 3 sources. The Amari distance was used as an accuracy measure. This analysis is also a contribution of this thesis.

6. The entire methodology of determining how many competing speakers can be accurately detected by human listeners along with experimental results is a personal contribution provided by the author of this thesis. This study can be considered novel. In order to demonstrate reliability of results the study was ran in two different sessions and the results were shown to correlate.

7. A custom software written in C# and running on Windows was developed in order to provide repeatability and relevance to the perception experiment discussed in the 6^{th} chapter. The software was developed entirely by the author of this thesis.

7.3 FUTURE WORK

As the personal contributions were summarized in a list we can do the same thing for the future research directions we are planning to follow:

1. In chapter 3.5 we demonstrated that neural networks can reduce the false classification ratio, when comparing to a pure threshold based classification. We consider that this study can be extended to using multiple types of neural networks with different parameter combinations in order to identify the maximum achievable performance.

2. In chapter 3.6, when a target input frame was aligned using DTW in time and frequency, an entire matrix of distances has resulted. Our approach was to sum the values to produce a single value metric. We consider that this can be studied deeper to produce an optimal scheme of combining distances. This can also have a positive impact on the amount of computational resources required.

3. Also, in chapter 3.6 we used only single speaker recordings as references. We believe that the performance of the methodology can be extended by using multi-speaker references as well and implement a voting based mechanism to select the correct number of speakers.

4. In order to reduce the experiments time we produced a fast native implementation of Dynamic Time Warping. We want to study if it is necessary to run DTW on the entire matrix or due to the nature of the compared symbols, a band of arbitrary length is necessary. This would reduce the complexity of the DTW and would set ground for even more extensive experimenting.

5. The results presented in chapter 5 that target the accuracy of the ICA based source separation algorithms, can be extended for more mixtures with varying competing speaker counts. Also, the experiments used instantaneous mixtures or modeled only an open space mixing case. This can be extended to use a room reverberation response because generally mixes are convolutional.

6. The perception experiment results presented in chapter 6 can be extended by using more volunteers of different ages and setting a finer control on the response time. Although the results correlated between the two sessions, this would add greater scientific significance to the research.

LIST OF PUBLICATIONS

[1]. [Andrei, 2011] V. Andrei, C. Paleologu, C. Burileanu, "*Implementation of a real-time reconfigurable text dependent speaker identification system*", to 5th International Conference on Speech Technology and Human-Computer Dialogue (SPED 2011), Vol. 1, pp. 1 – 6, 2011

[2]. [Andrei, 2014] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "*Detecting the number of competing speakers – human selective hearing versus spectrogram distance based estimator*", Proceedings of 15th Annual Conference of the International Speech Communication Association (Interspeech 2014), pp. 467 – 470, 2014

[3]. [Andrei, 2014b] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "*Perception study inspired method for automatically detecting the number of competing speakers*", University "Politehnica" of Bucharest Scientific Bulletin, Vol. 3C, 2014, ISSN 1223-7027

[4]. [Andrei, 2015] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "*Counting competing speakers in a timeframe – human versus computer*", Proceedings of 16th Annual Conference of the International Speech Communication Association (Interspeech 2015),

[5]. [Andrei, 2015b] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "*Estimating competing speaker count for blind speech source separation*", Proceedings of 8th International Conference on Speech Technology and Human-Computer Dialogue (SPED 2015)

[6]. [Andrei, 2016] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "*Analysis of speaker count Discrimination strategies for single channel crowd-sensing*", **submitted** to 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)

REFERENCES

[Nicolau, 1963] E. Nicolau, I. Wever, St. Gavat, "Aparat pentru recunoaştera automată a vocalelor", Automatica și Electronica, (6), 1963

[Marquardt, 1963] D.W. Marquardt. An Algorithm for the Least-Squares Estimation of Nonlinear Parameters. SIAM Journal of Applied Mathematics, 11(2):431–441, Jun. 1963

[Welch, 1967] P. D. Welch, "The use of Fast Fourier Transform for estimation of power spectra: a method based on time averaging over short modified periodograms", IEEE Transactions on Audio Electroacoustics, AU-15, pp. 70-73, 1967

[Itakura, 1970] F. Itakura, S. Saito, "A statistical method for estimation of speech spectral density and formant frequencies", Electronics & Communications in Japan, 53A, pp. 36-43, 1970

[Itakura, 1975] F. Itakura, "Minimum prediction residual principle applied to speech recognition", IEEE ASSP-23, pp. 62-72, 1975

[Gray, 1976] A. H. Gray, J. D. Markel, "Distance measures for speech processing", IEEE ASSP-24(5): pp. 380-391, 1976

[Davis, 1980] S. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. ASSP, vol. 28, no. 4, pp. 357–366, 1980

[Burileanu, 1983] Burileanu, C., Teodorescu, V., Stolojanu, G., Radu, C., "Sistem cu logică programată pentru recunoaşterea cuvintelor izolate, independent de vorbitor," Artificial intelligence and robotics, Romanian Academy Publishing House, Bucharest, vol. 1, pp.266-274, 1983

[Groza, 1984] V. Groza, M. Boldea, C. Bărbulescu, "Recunoaşterea vocalelor cu ajutorul unui microsistem de calcul", Buletinul sesiunii științifice pentru tineret "Tehnic 2000", pag. 274 – 77, Timișoara, aprilie 1984

[Huber, 1985] P. Hubber, "Projection pursuit", The annals of statistics, 13(2):435-475

[Furui, 1986] Furui, S., "Speaker-independent isolated word recognition using dynamic features of speech spectrum", IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 34, no. 1, pp. 52-59., 1986

[Jones, 1987] M. Jones, R. Sibson, "What is projection pursuit?" Journal of the Royal Statistical Society, Ser. A, 150:1-36, 1987

[Lee, 1988] K. F. Lee, "The CMU Sphinx System", Ph.D. Thesis, CMU, 1988

[Rabiner, 1989], Rabiner L. R., "A tutorial on hidden Markov models and selected applications in speech recognition", Proceedings of the IEEE,. Vol. 77, No. 2., pp. 257-285, 1989

[Hermansky, 1990] H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," Journal Acoust. Soc. Amer., vol. 87, no. 4, pp. 1738–1752, 1990

[Kuhn, 1990] R. Kuhn, R. De Mori, "A cache based natural language model for speech recognition", In IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 12, No. 6, pp. 570-583, 1990

[AN4, 1991] CMU Robust Speech Recognition Group: Census Database

[Cover, 1991] T.M. Cover, J.A. Thomas, "Elements of information theory", Wiley, 1991

[Pham, 1992] D.-T.Pham, J. Joutsenalo, "Separation of a mixture of independent sources through a maximum likelihood approach". In Proc. EUSIPCO, pages 771–774.

[Rabiner, 1993], Rabiner L. R., Juang B. H., "Fundamentals of speech recognition", Prentice Hall. 1993

[TIMIT, 1993] J. Garolfo, L. Lamel, et. al., "TIMIT Acoustic-Phonetic Continuous Speech Corpus"

[Belouchrani, 1993] A. Belouchrani, K. Abed-Meraim, J. F. Cardoso, E. Moulines, "Second-order blind separation of temporally correlated sources", Proc. of Intl. Conf. on Digital Signal Processing, pp. 346 – 351, 1993

[Robinson, 1994] A. Robinson, "An application to recurrent nets to phone probability estimation," IEEE Trans. Neural Networks, vol. 5, no. 2, pp. 298–305, 1994.

[Amari, 1996] Amari, S., Cichocki, A. and Yang, H.H., "A new learning algorithm for blind signal separation", Advances in Neural Information Processing Systems, **8**, 757—763, 1996

[Ming, 1998] J. Ming and F. J. Smith, "Improved phone recognition using Bayesian triphone models," in Proc. ICASSP, 1998, pp. 409–412, 1998

[Halberstadt, 1998] A. Halberstadt and J. Glass, "Heterogeneous measurements and multiple classifiers for speech recognition," in Proc. ICSLP, 1998

[Burileanu, 1998] Burileanu, D., Sima, M., Burileanu, C., Croitoru, V., "A Neuronal Network-Based Speaker-Independent System for Word Recognition in Romanian Language," TSD 1998, pp. 177-182, 1998

[Grigore, 1998] Grigore, O., Gavăt, I. and Zirra, M., "Neural network vowel recognition in Romanian language," CONTI 1998, pp. 165-172, Timişoara, Romania, 1998

[Valsan, 1998b] Valsan, Z., Sabac, B., Gavăt, I., "Combining self organizing feature map and multilayer perceptron in a neural system for fast key-word spotting," SPECOM 1998, pp. 303-308, St. Petersburg, Russia, 1998

[Hyvarinen, 1998] A. Hyvarinen, "New approximations of differential entropy for independent component analysis and projection pursuit", Advances in neural information processing systems, volume 10, pages 273-279, MIT Press, 1998

[Fung, 1999] P. Fung, L. Kat, "Fast accent identification and accented speech recognition", Proc. of ICASSP, pp. 221-224, 1999

[Hyvarinen, 1999] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis", IEEE Transactions on Neural Networks, Vol. 10, Issue 3, pp. 626-634

[Hyvarinen, 2000] A. Hyvarinen, E. Oja, "Independent Component Analysis: Algorithms and Applications", Neural Networks, Vol. 13, pp. 411-430, 2000

[Nuzillard, 2000] D. Nuzillard and A. Bijaoui, "Blind source separation and analysis of multispectral astronomical images," Astron. Astrophys. Suppl. Ser., vol. 147, pp. 129–138, Nov. 2000

[FastICA, 2000] http://research.ics.aalto.fi/ica/fastica/

[Choi, 2000] S. Choi, A. Cichocki and S. Amari, "Flexible independent component analysis," Journal of VLSI Signal Processing, 26, 25-38, Aug. 2000

[Jelinek, 2001], Jelinek F., "Statistical Methods for speech recognition", The MIT Press, 2001.

[Reynolds, 2001], D. Reynolds, "Gaussian Mixture Models", MIT Lincoln Laboratory, 2001

[Ghahramani, 2001] Z. Ghahramani, "An introduction to hidden Markov models and Bayesian Networks", International Journal of Pattern Recognition and Artificial Intellicence, pp. 9-42, 2001

[Pfau, 2001] T. Pfau, D. Ellis, A. Stolcke, "Multispeaker speech activity detection for the ICSI Meeting Recorder", Proc. ASRU 2001

[Renevey, 2001] P. Renevey, A. Drygajlo, "Entropy based voice activity detection in very noisy conditions", 2nd Interspeech Conference, 2011

[Prasad, 2002] R. V. Prasad, A. Sangwan, H. S. Jamadagni, et. al. "Comparison of Voice Activity Detection Algorithms for VoIP", Proceedings of the 7th International Symposium on Computers and Communications, 2002

[Dong, 2002] E. Dong, G. Liu, Y. Zhou, and X. Zhang. Applying Support Vector Machines to Voice Activity Detection. In Proceedings of the International Conference on Signal Processing (ICSP), 2002

[Raphael, 2002] C. Raphael, "Automatic transcription of piano music," in Proc. ISMIR, 2002

[Brookes, 2002] M. Brookes, "Matlab Toolbox for Speech Processing", Department of Electronics and Engineering Imperial, 2002

[Anemuller, 2003] J. Anemuller, T. J. Sejnowski, and S. Makeig, "Complex independent component analysis of frequency-domain electroencephalographic data," Neural Networks, vol. 16, no. 9, pp. 1311–1323, Nov 2003

[Sakuraba, 2003] Y. Sakuraba and H. Okuno, "Note recognition of polyphonic music by using timbre similarity and direction proximity," in Proc. ICMC, 2003

[Samaragdis, 2003] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," in Proc. WASPAA, 2003

[Vreeken, 2003] J. Vreeken, "Spiking neural networks – an introduction", Adaptive Intelligence Laboratory, Intelligent Systems Group

[Tchorz, 2003] J. Tchorz and B. Kollmeier, "Snr estimation based on amplitude modulation analysis with applications to noise suppression," IEEE Trans. Speech, Audio Process., vol. 11, no. 3, pp. 184–192, May 2003

[Gustafsson, 2003] F. Gustafsson, F. Gunnarson, "Positioning using time difference of arrival measurements", Proc. of ICASSP 2003

[Arai, 2003] T. Arai, "Estimating number of speakers by the modulation characteristics of speech", Proc. of ICASSP 2003, Vol. 2, pp. 197 – 200.

[Cichoki, 2003] A. Cichocki, S. Amari, "Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications", Wiley, 2003

[Miller, 2003] E. G. Learned-Miller, J. W. Fisher, "ICA using spacing estimates on entropy", Journal of Machine Learning Research, Vol. 4, pp. 1271 – 1295, 2003

[Oancea, 2004] Oancea, E., Gavăt, I., Dumitru, O., Munteanu, D., "Continuous Speech Recognition for Romanian Language Based on Context Dependent Modeling," Communications 2004, Bucharest, pp. 221-224, 2004

[Vincent, 2004] E. Vincent, "Modeles ` d'instruments pour la separation ´ de sources et la transcription d'enregistrements musicaux," Ph.D. dissertation, IRCAM, 2004

[Ramirez, 2004] J. Ramirez, J. C. Segura, C. Benitez, et. al., "Efficient voice activity detection algorithms using long-term speech information", Elsevier Speech Communication, Vol. 42, pp. 271-287

[Roth, 2005] Roth G, Dicke U (May 2005). "Evolution of the brain and intelligence". Trends Cogn. Sci. (Regul. Ed.) 9 (5): 250–257, 2005

[Honig, 2005] F. Honig, G. Stemmer, C. Hacker, F. Brugnare, "Revising Perceptual Linear Prediction", Proc. of Interspeech 2005, pp. 2997-3000

[Verstraeten, 2005] D. Verstraeten, B. Schrauwen, D. Stroobandt, "Isolated word recognition using a liquid state machine", In ESANN 2005, European Symposium on Artificial Neural Network, pp. 435-440

[Wang, 2005] G. Wang, M. Pavel, "A spiking neuron representation of auditory signals", In International Joint Conference on Neural Networks, pp. 416-421

[Loiselle, 2005] S. Loiselle, J. Rouat, D. Pressnitzer, S. Thorpe, "Exploration of rank order coding with spiking neural networks for speech recognition", In International Joint Conference on Neural Networks, pp. 2076-2080

[Zheng, 2005], Y. Zheng, R. Sproat, et. al., "Accent detection and speech recognition for Shanghai-accented Mandarin", nlp.standford.edu

[Ellis, 2005] D. P. W. Ellis, "PLP and RASTA (and MFCC, and Inversion) in Matlab," 2005 [Online]. Available: http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/

[Vogt, 2005] T. Vogt, E. Andre, N. Bee, "EmoVoice – A framework for online recognition of emotions from voice", Proceedings of the 4th IEEE tutorial and research workshop on Perception and Interactive Technologies for Speech-Based Systems: Perception in Multimodal Dialogue Systems

[Brown, 2005] Wang, L and Brown G. J., "Computational Auditory Scene Analysis", John Wiley & Sons, ISBN 0-471-45435-4, 2005

[Coch, 2005] Coch, D., Sanders, L. D., Neville, H. J., "An event related potential study of selective auditory attention in children and adults", Journal of Cognitive Neuroscience, Vol 17, Nr. 4, 2005

[Cristea, 2006] Cristea, D., Forăscu, C., "Linguistic Resources and Technologies for Romanian Language", Journal of Computer Science of Moldova, vol. 14, no. 1, pp. 33-73, 2006

[Voxtec, 2006] Phraselator Device - http://www.voxtec.com/phraselator/

[Mansour, 2006] A. Mansour, N. Benchekroun, and C. Gervaise, "Blind separation of underwater acoustic signals," in ICA'06, 2006, pp. 181–188.

[Vaya, 2006] C. Vaya, J. J. Rieta, C. S´anchez, and D. Moratal, "Performance study of convolutive BSS algorithms applied to the electrocardiogram of atrial fibrillation," in ICA'06, 2006, pp. 495–502

[Choueiter, 2006] G. Choueiter, D. Povey, et. al., "Morpheme based language modeling for Arabic LVCSR", Proc. of ICASSP 2006, vol 1., pp. 1053-1057

[Maraboina, 2006] S. Maraboina, D. Kolossa, P. K. Bora, R. Orglmeister, "Multi-speaker voice activity detection using ICA and beampattern analysis", 14th European Signal Processing Conference EUSIPCO 2006

[Nilsson, 2006] M. Nillson, "Entropy and Speech", Thesis for the degree of Doctor of Philosophy, Sound and Image Processing Laboratory, School of Electrical Engineering, KTH, 2006

[Makino, 2007] S. Makino, T.-W. Lee, H. Sawada, "Blind Speech Separation", ISBN 978-1-4020-6478-4

[Pedersen, 2007] M. S. Pedersen, J. Larsen, U. Kjems, L. C. Parra, "A survey of convolutive blind source separation methods", Springer handbook of speech processing, November 2007, ISBN 978-3-540-49125-5

[Brockman, 2007] J. Brockman, "Recursion and Human Thought – Why the piraha do not have numbers", www.edge.org, 2007

[Mporas, 2007] I. Mporas, T. Ganchev, M. Siafarikas, N. Fakotakis, "Comparison of speech features on the speech recognition task", Journal of Computer Science 2007, pp. 608 – 616

[Gales, 2007] M. Gales, S. Young, "The Application of Hidden Markov Models in Speech Recognition", Foundation and Trends in Signal Processing, Vol. 1., No. 3, 2007

[Deng, 2007] L. Deng and D. Yu, "Use of differential cepstra as acoustic features in hidden trajectory modelling for phonetic recognition," in Proc. ICASSP, 2007, pp. 445–448

[Saraswathi, 2007] S. Saraswathi, T. Geetha, "Morpheme based language model for Tamil speech recognition system", The International Arab Journal of Information Technology, Vol. 4, No. 3, pp. 214-220

[Kinnunen, 2007] T. Kinnunen, E. Chernenko, M. Tuononen, P. Franti, and H. Li. Voice activity detection using MFCC features and support vector machine. In Proceedings of the International Conference on Speech and Computer, 2007

[Rivet, 2007] B. Rivet, L. Girin, C. Jutten, "Visual voice activity detection as a help for speech source separation from convolutive mixtures", Elsevier Speech Communication, vol. 49, pp. 667-677, 2007

[Douglas, 2007] S. C. Douglas, M. Gupta, H. Sawada, S. Makino, "Spatio-temporal FastICA algorithms for the blind separation of convolutive mixtures", IEEE Transactions on Audio, Speech and Language Processing, Vol. 15, pp. 1511 – 1520, 2007

[Petrea, 2008] Petrea, C., Ghelmez-Hanes, D., Popescu, V., Buzo, A., Burileanu,, C., "Spontaneus Speech Database for Romanian Language," ECIT 2008, Iaşi, Romania, 2008

[Havelock, 2008] D. Havelock, S. Kuwano, M. Vorlander, "Handbook of Signal Processing in Acoustics", vol. 1, Springer, pp. 486

[Hastie, 2008], Hastie, Tibshirani, "Gaussian mixture models", SL&DM, standford.edu, November, 2008

[Almajai, 2008] I. Almajai, B. Milner, "Using audio-visual features for robust voice activity detection in clean and noisy speech", IEEE 16th European Signal Processing Conference, pp. 1-5, 2008

[Lehman, 2008] E. Lehmann, A. Johansson, "Prediction of energy decay in room impulse response simulated with an image source model", Journal of the Acoustical Society of America, Vol. 124, pp. 269-277, 2008

[Marquard, 2009] Stephen Marquard, 2009 – "Sphinx4 speech recognition results for selected lectures from Open Yale Courses", http://trulymadlywordly.blogspot.ro/2011/12/sphinx4-speech-recognition-results-for.html

[Militaru, 2009] Militaru, D., Gavăt, I., Dumitru, O., Zaharia, T., Segărceanu, S., "Protologos, System for Romanian Language Automatic Speech Recognition and Understanding," SpeD 2009, pp. 21 – 32, Constanţa, Romania, 2009

[Hifny, 2009] Y. Hifny and S. Renals, "Speech recognition using augmented conditional random fields," IEEE Trans. Audio Speech Lang. Processing, vol. 17, no. 2, pp. 354–365, 2009

[Moisy, 2009] H. P. Moisy, S. Bohte, "Computing with spiking neuron networks"

[Kim, 2009] G. Kim, Y. Lu, Y. Hu, and P. C. Loizou, "An algorithm that improves speech intelligibility in noise for normal-hearing listeners," J. Acoust. Soc. Amer., vol. 126, pp. 1486–1494, 2009

[Werner, 2009] Werner, L., "Development of Auditory Behavior: Hearing Science", University of Washington Course, 2009

[Kokkinakis, 2010] K. Kokkinakis, P. C. Loizou, "Advances in modern blind signal separation algorithms: Theory and applications", ISSN 1938-1727

[Perfors, 2010] A. Perfors, J. Tanenbaum, E. Gibson, T. Regier, "How recursive is a language? A Bayesian exploration" http://tedlab.mit.edu, 2010

[Shrawankar, 2010] U. Shrawankar, V. Thakare, "Techniques for feature extraction in speech recognition systems: a comparative study", SGB Amravati University, 2010

[Hinton, 2010] G. Hinton, "A practical guide to training restricted Boltzmann machines", Department of Computer Science, University of Toronto, 2010

[Mohamed, 2010] A. Mohamed, D. Yu, and L. Deng, "Investigation of full-sequence training of deep belief networks for speech recognition," in Proc. Interspeech, 2010, pp. 2846–2849

[Dahl, 2010] G. E. Dahl, M. Ranzato, A. Mohamed, and G. E. Hinton, "Phone recognition with the mean-covariance restricted Boltzmann machine," in Advances in Neural Information Processing Systems 23, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, Eds. Cambridge, MA: MIT Press, 2010, pp. 469–477

[Tachbelle, 2010] M. Y. Tachbelle, W. Menzel, "Capturing word-level dependencies in morpheme-based language modeling", Proceedings of 2nd Workshop on African Language Technology, 2010

[Shin, 2010] J. W. Shin, J-H. Chang, N. S. Kim, "Voice activity detection based on statistical models and machine learning approaches", Elsevier Computer Speech and Language, Vol. 24, pp. 515 – 530

[Sayoud, 2010] H. Sayoud, S. Ouamour, "Proposal of a new confidence parameter estimating the number of speakers – an experimental investigation", Journal of Information Hiding and Multimedia Signal Processing, Vol. 1, 2010

[Mactech, 2010] www.mactech.com, "Apple Patent Involves Audio BeamForming", May 2010

[Microsoft, 2011] research.microsoft.com - http://research.microsoft.com/en-us/people/gokhant/jamia11.pdf

[PC World, 2011] www.pcworld.com – "IBM Watson Vanquishes Human Jeopardy Foes" - http://www.pcworld.com/article/219893/ibm_watson_vanquishes_human_jeopardy_foes.html

[Cucu, 2011] Horia Cucu, "Towards a speaker-independent, large-vocabulary continuous speech recognition system for Romanian", Teză de Doctorat, Universitatea Politehnica București

[Puigt, 2011] M. Puigt, "A very short introduction to blind source separation", Foundation for research and technology – Hellas, Institute of computer science, April 12 / May 3 – 2011

[Andrei, 2011] V. Andrei, C. Paleologu, C. Burileanu, "Implementation of a real-time reconfigurable text dependent speaker identification system", Proc. of SPED 2011

[Sainath, 2011] T. N. Sainath, B. Ramabhadran, M. Picheny, D. Nahamoo, and D. Kanevsky, "Exemplar-based sparse representation features: From TIMIT to LVCSR," IEEE Trans. Audio Speech Lang. Processing, vol. 19, no. 8, pp. 2598–2613, Nov. 2011

[Mousa, 2011] A. E.D. Mousa, M. A. B. Shaik, R. Schulter, H. Ney, "Morpheme based factored language models for German LVCSR", Proc. of Interspeech 2011, pp. 1445-1449

[Biadsy, 2011], F. Biadsy, "Automatic dialect and accent recognition and its application to speech recognition", Ph.D Thesis, Columbia University, 2011

[Ayadi, 2011], M. E. Ayadi, M. S. Kamel, "Survey on speech emotion recognition: Features, classification, schemes, and databases", Pattern Recognition, Vol. 44, Issue 3, pp. 572-587

[Gomes, 2011] Gomes, H., Duff, M., Ramos. M, Molholm, S, Foxe, J., Halperin, J., "Auditory selective attention and processing in children with attention deficit/hyperactivity disorder", Journal of Clinical Neurophysiology, August 2011

[CNN, 2012] www.cnn.com – Technology / Fortune - http://tech.fortune.cnn.com/2012/06/29/minneapolis-street-test-google-gets-a-b-apples-siri-gets-a-d/

[Alam, 2012] J. Alam, T. Kinnunen, etc. "Multitaper MFCC and PLP features for speaker verification using i-vectors", School of Computing – University of Eastern Finland, 2012

[Mohamed, 2012] A. Mohamed, G. E. Dahl, G. Hinton, "Acoustic modeling using deep belief networks", IEEE Transactions on Audio, Speech and Language Processing, 2012

[CNN, 2012] www.cnn.com – Technology / Fortune - http://tech.fortune.cnn.com/2012/06/29/minneapolis-street-test-google-gets-a-b-apples-siri-gets-a-d/

[Dahl, 2012] G. E. Dahl, D. Yu, L. Deng, A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition". IEEE Transactions on Audio, Speech and Language Processing, 2012

[Hinton, 2012] G. Hinton, L. Deng, G. Dahl, A. Mohamed, et. al., "Deep neural networks for acoustic modeling in speech recognition", IEEE Signal Processing Magazine, November 2012

[Hamid, 2012] O. Abdel-Hamid, A. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in Proc. ICASSP, 2012, pp. 4277–4280

[Yu, 2012] D. Yu, S. Sinislachi, L. Deng, C. Lee, "Boosting attribute and phone estimation accuracies with deep neural networks for detection-based speech recognition" in Proc. ICASSP, 2012, pp. 4169-4172

[Chen, 2012] G. Chen, K. Kumatani, J. McDonough, B. Raj, "Distant multi-speaker voice activity detection using relative energy ratio", Proc. of ICASSP 2012

[ITU, 2012] International Telecommunication Union, http://www.itu.int/rec/T-REC-P.56-201112-I/en

[Han, 2013], K. Han, "Emotion detection from speech signals", Microsoft Research, 2013

[Pal, 2013] A. R. Pal, D. Saha, "Detection of slang words in e-data using semi supervised learning", International Journal of Artificial Intelligence & Applications, Vol. 4, No. 5, pp. 49-61

[Zhang, 2013] X.L. Zhang, J. Wu, "Deep belief networks based voice activity detection", IEEE Transactions on Audio, Speech, And Language Processing, Vol. 21, No. 4, pp. 697 – 710, 2013

[Germain, 2013] F. G. Germain, D. L. Sun, G. J. Mysore, "Speaker and Noise Independent Voice Activity Detection", Proc. of Interspeech 2013

[Minotto, 2013] V. P. Minotto, "Audiovisual voice activity detection and localization of simultaneous speech sources", PhD. Thesis – Universidade Federal Do Rio Grande Do Sul, 2013

[Wikibooks, 2014] http://en.wikibooks.org/wiki/ Engineering_Acoustics/ Outdoor Sound Propagation Engineering Acoustics / Outdoor sound propagation

[Pereltsvaig, 2014] A. Pereltsvaig, "What is phonemic diversity? – And does it prove the "Out of Africa" theory?" www.languagesoftheworld.info, May 24, 2014

[Minotto, 2014] V. P. Minotto, C. R. Jung, B. Lee, "Simultaneous-speaker voice activity detection and localization using mid-fusion of SVM and HMMs", IEEE Transactions on Multimedia, pp. 1032-1044

[Andrei, 2014] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "Detecting the number of competing speakers – human selective hearing versus spectrogram distance based estimator," Proceedings of Interspeech 2014, pp. 467 – 470, 2014

[Interspeech, 2015] "Speech beyond speech: towards a better understanding of the most important biosignal", http://interspeech2015.org/papers/scientific-areas/

[Andrei, 2015] V. Andrei, H. Cucu, A. Buzo, C. Burileanu, "Counting competing speakers in a timeframe – human versus computer", Proceedings of Interspeech 2015.